

CA Process Automation

Referência da API de serviços web

Release 04.2.00



A presente documentação, que inclui os sistemas de ajuda incorporados e os materiais distribuídos eletronicamente (doravante denominada Documentação), destina-se apenas a fins informativos e está sujeita a alterações ou remoção por parte da CA a qualquer momento. Esta Documentação contém informações proprietárias da CA e não pode ser copiada, transferida, reproduzida, divulgada, modificada nem duplicada, parcial ou completamente, sem o prévio consentimento por escrito da CA.

Se o Cliente for um usuário licenciado do(s) produto(s) de software referido(s) na Documentação, é permitido que ele imprima ou, de outro modo, disponibilize uma quantidade razoável de cópias da Documentação para uso interno seu e de seus funcionários envolvidos com o software em questão, contanto que todos os avisos de direitos autorais e legendas da CA estejam presentes em cada cópia reproduzida.

O direito à impressão ou, de outro modo, à disponibilidade de cópias da Documentação está limitado ao período em que a licença aplicável ao referido software permanecer em pleno vigor e efeito. Em caso de término da licença, por qualquer motivo, fica o usuário responsável por garantir à CA, por escrito, que todas as cópias, parciais ou integrais, da Documentação sejam devolvidas à CA ou destruídas.

NA MEDIDA EM QUE PERMITIDO PELA LEI APLICÁVEL, A CA FORNECE ESTA DOCUMENTAÇÃO "NO ESTADO EM QUE SE ENCONTRA", SEM NENHUM TIPO DE GARANTIA, INCLUINDO, ENTRE OUTROS, QUAISQUER GARANTIAS IMPLÍCITAS DE COMERCIALIZABILIDADE, ADEQUAÇÃO A UM DETERMINADO FIM OU NÃO VIOLAÇÃO. EM NENHUMA OCASIÃO, A CA SERÁ RESPONSÁVEL PERANTE O USUÁRIO OU TERCEIROS POR QUAISQUER PERDAS OU DANOS, DIRETOS OU INDIRETOS, RESULTANTES DO USO DA DOCUMENTAÇÃO, INCLUINDO, ENTRE OUTROS, LUCROS CESSANTES, PERDA DE INVESTIMENTO, INTERRUPÇÃO DOS NEGÓCIOS, FUNDO DE COMÉRCIO OU PERDA DE DADOS, MESMO QUE A CA TENHA SIDO EXPRESSAMENTE ADVERTIDA SOBRE A POSSIBILIDADE DE TAIS PERDAS E DANOS.

O uso de qualquer software mencionado na Documentação é regido pelo contrato de licença aplicável, e tal contrato não deve ser modificado de nenhum modo pelos termos deste aviso.

O fabricante desta Documentação é a CA.

Fornecida com "Direitos restritos". O uso, duplicação ou divulgação pelo governo dos Estados Unidos está sujeita às restrições descritas no FAR, seções 12.212, 52.227-14 e 52.227-19(c)(1) - (2) e DFARS, seção 252.227-7014(b)(3), conforme aplicável, ou sucessores.

Copyright © 2010 CA. Todos os direitos reservados. Todas as marcas comerciais, nomes de marcas, marcas de serviço e logotipos aqui mencionados pertencem às suas respectivas empresas.

Referências a produtos da CA Technologies

Este documento faz referência aos seguintes produtos da CA Technologies:

- CA Catalyst para o CA Service Desk Manager (CA Catalyst Conector do CA SDM)
- CA Client Automation (anteriormente CA IT Client Manager)
- CA Configuration Automation (anteriormente, CA Cohesion® Application Configuration Manager)
- CA CMDB
- CA eHealth®
- CA EEM (Embedded Entitlements Manager)
- CA Infrastructure Insight (anteriormente Bundle: CA Spectrum IM & CA NetQoS Reporter Analyzer combinados)
- CA NSM
- CA Process Automation (anteriormente, CA IT Process Automation Manager)
- CA Service Catalog
- CA SDM (CA Service Desk Manager)
- CA Service Operations Insight (CA SOI) (anteriormente CA Spectrum® Service Assurance)
- CA SiteMinder®
- CA Workload Automation AE

Entrar em contato com o Suporte técnico

Para assistência técnica online e uma lista completa dos locais, principais horários de atendimento e números de telefone, entre em contato com o Suporte técnico pelo endereço <http://www.ca.com/worldwide>.

Índice

Capítulo 1: Introdução aos serviços web 9

Capítulo 2: Referência da API RESTful 11

Serviços web RESTful e CA Process Automation	11
Catalyst Process Automation Services	12
Catalyst REST	51
Utilização de OData do Catalyst	106

Capítulo 3: Referência da API SOAP 109

Métodos de serviços web	109
AsyncSoapResponse	109
checkServerStatus	110
checkStartRequestStatus	111
controlInstance	113
controlProcess	115
deleteArchivedInstances	118
deleteAttachments	119
executePendingInteraction	120
executeProcess	123
executeStartRequest	126
exportObject	130
generateEvent	133
getAttachments	136
getITPamVersionInfo	138
getMatchingEvents	140
getPendingInteractionRequestForm	142
getPendingUserInteractions	146
getProcessLogs	152
getProcessStatus	159
getStartRequestForm	162
getStartRequestForms	164
ImportObject	168
Tags comuns para métodos de serviços web	172
As tags <page> e <pages>	173
A tag <attachments>	175
A tag <params>	176

A tag <options>	176
Apêndice A: Status do HTTP e códigos de erro	177
Status do HTTP e códigos de erro	178
Códigos de Status conhecidos	180

Alterações na documentação

Foram feitas as seguintes atualizações de documentação desde a última release desta documentação:

- [Nome do tópico]-[Descrição do novo capítulo, tópico ou descrição de alterações em um tópico existente].
- [Nome do tópico]-[Descrição do novo capítulo, tópico ou descrição de alterações em um tópico existente].

Capítulo 1: Introdução aos serviços web

Os serviços web são APIs (application programming interfaces, interfaces de programação de aplicativos) que fornecem aos usuários a capacidade de se comunicar com o CA Process Automation de fontes externas.

O CA Process Automation fornece dois métodos para usar os serviços web: REST e SOAP. Essa referência descreve as várias APIs que você pode usar para se comunicar por meio de ambos os métodos.

Capítulo 2: Referência da API RESTful

REST é uma alternativa para os serviços web baseados em SOAP. Ao usar o Catalyst Process Automation Services, o CA Process Automation pode ser acessado por uma operação sem periféricos por meio de serviços Catalyst RESTful.

Esta seção contém os seguintes tópicos:

[Serviços web RESTful e CA Process Automation](#) (na página 11)

Serviços web RESTful e CA Process Automation

REST é um método de design que é com base em HTTP. REST permite que os serviços web identifiquem e manipulem os recursos.

A implementação de serviços RESTful segue estes princípios de criação:

- Usar métodos HTTP explicitamente
- Não ter monitoramento de estado
- Acessar recursos como os URIs
- Oferecer suporte a XML e JSON (JavaScript Object Notation)

O estilo arquitetural que do RESTful obteve está sendo amplamente adotado na web. O CA Process Automation expõe os serviços RESTful por meio do Catalyst Process Automation Services, que a tecnologia do cliente RESTful pode consumir. Os criadores podem usar qualquer cliente REST para acessar o CA Process Automation no modo sem periférico, o qual permite a configuração e o controle do CA Process Automation sem usar a interface do usuário.

Este capítulo descreve as solicitações RESTful que um criador pode enviar Catalyst Process Automation Services para interagir com o CA Process Automation.

Para obter mais informações, consulte a seguinte documentação:

- Os [tópicos](#) (na página 12) do Catalyst Process Automation Services para obter informações sobre como executar os serviços web RESTful com os operadores do CA Process Automation.
- O *Guia do Administrador de Conteúdo* para obter informações sobre como implantar o Catalyst Process Automation Services.
- O *Guia de Referência do Criador de Conteúdo* para obter mais informações sobre os operadores do Catalyst.

Catalyst Process Automation Services

O recipiente do Catalyst e o Catalyst Process Automation Services estão incorporados ao CA Process Automation. A comunicação é otimizada entre o recipiente do Catalyst, o Catalyst Process Automation Services e o CA Process Automation.

O Catalyst Process Automation Services inclui um conector integrado que expõe as interfaces do UCF. Qualquer aplicativo cliente UCF (incluindo os operadores do CA Process Automation Catalyst) pode utilizar as interfaces do UCF.

Qualquer cliente Catalyst pode utilizar o conector do CA Process Automation para executar as seguintes ações:

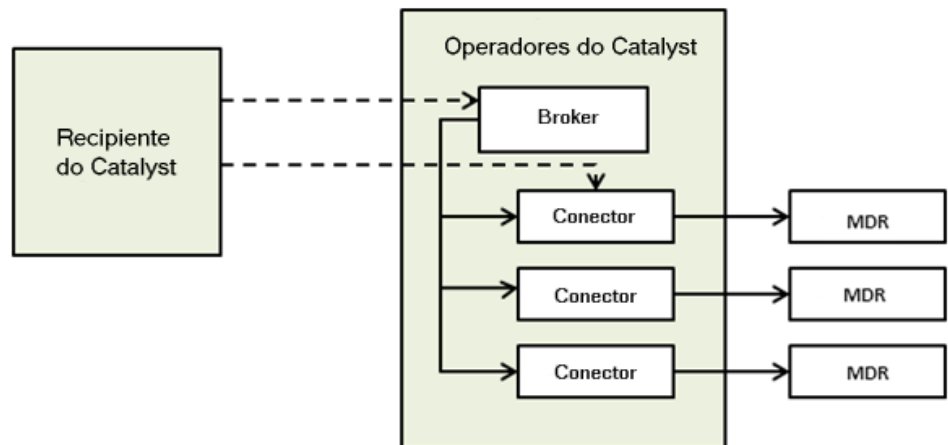
- Consultar o status de processos do CA Process Automation.
- Iniciar, cancelar, suspender e retomar um processo do CA Process Automation.
- Consultar e assinar alertas.
- Consultar e enviar formulários de solicitação inicial.
- Retirar da fila e encerrar solicitações iniciais.
- Consultar, atualizar e responder às tarefas (solicitações de interação).
- Importar e exportar objetos de automação.
- Receber eventos de ciclo de vida para processos do CA Process Automation.
- Consultar e atualizar conjuntos de dados.
- Consultar e atualizar módulos do CA Process Automation.

Importante: não instale outros conectores no recipiente do Catalyst Process Automation.

Uso:

Qualquer interface de cliente do Catalyst, incluindo a interface do Catalyst REST e os operadores do Catalyst que o CA Process Automation inclui, permite acessar o Catalyst Process Automation Services.

Todos os operadores do Catalyst interagem com conectores do Catalyst, que interagem com um MDR (Management Data Repository - Repositório de Dados de Gerenciamento). Um conector Catalyst sempre reside em um recipiente do Catalyst. O recipiente do Catalyst tem um broker (um serviço de pesquisa de nós) que publica as configuração de nós de cada conector no recipiente.



Todos os operadores do Catalyst exigem o URL do broker do Catalyst e a `MdrProdInstance` do conector. Quando você digita um URL do broker do Catalyst em um operador do Catalyst, o operador consulta o broker sobre as configurações de nós. Em seguida, o broker preenche a `MdrProdInstance` com uma lista dessas configurações de nós.

Depois que o URL do broker do Catalyst e a `MdrProdInstance` estiverem definidos em um operador do Catalyst, você poderá especificar as operações REST a serem executadas.

Comunicações

O Catalyst Process Automation Services requer comunicação segura do Catalyst. O URL do broker do Catalyst Process Automation Services é:

```
https://<nome_do_host>:7443/ucf/BrokerService
```

O broker seguro exige credenciais do Catalyst. Se o broker seguro estiver definido na configuração OasisConfig.properties "use.catalyst.claims.credentials", ele também exigirá credenciais do CA Process Automation. Você pode especificar essas credenciais na seção de Segurança do UCF de qualquer operador Catalyst. Os valores de credencial são inseridos nos parâmetros de segurança do Catalyst de um operador. Os nomes das reivindicações são Nome de usuário e Senha.

Operadores Catalyst

Para atingir os seguintes objetivos, crie processos do CA Process Automation que usem os operadores do Catalyst:

- Consultar o conector (operador Obter)
- Chamar as operações do conector (operador Execute)
- Assinar as alterações do conector (operador Inscrever-se para receber alterações)
- Consultar o status dos processos do CA Process Automation
- Iniciar, cancelar, suspender e retomar um processo do CA Process Automation.
- Consultar e assinar alertas.
- Consultar e enviar formulários de solicitação inicial.
- Retirar da fila e encerrar solicitações iniciais.
- Consultar, atualizar e responder às tarefas (solicitações de interação).
- Importar e exportar objetos de automação.
- Receber eventos de ciclo de vida relacionados a processos do CA Process Automation.
- Consultar e atualizar conjuntos de dados.
- Consultar e atualizar módulos do CA Process Automation.

Observação: o arquivo ucfpamconnector-descriptors.jar contém descritores de conectores do CA Process Automation.

Parâmetros de segurança do Catalyst

Todos os operadores do Catalyst incluem parâmetros de Segurança do Catalyst que oferecem suporte à autenticação em nível de Catalyst e de conector.

Após obter acesso aos nós do Catalyst, você poderá usar requisições para obter informações de segurança específicas do conector. Para obter informações sobre reivindicações do conector, consulte o guia Conector fornecido com o conector Catalyst aplicável.

Os administradores em nível de domínio definem as configurações de segurança padrão para a categoria de operador do Catalyst (que contém *todos* os operadores do Catalyst). Essas configurações são herdadas. É possível editar essas configurações para operadores individuais nos níveis de ambiente, orquestrador e agente.

Consulte a *Referência de interface de usuário* para obter informações sobre os parâmetros de segurança em nível de categoria de operador (ou de módulo). Consulte a *Referência do criador de conteúdo* para obter informações sobre os parâmetros de segurança para cada operador.

Executar parâmetros

Cada operador do Catalyst requer parâmetros de entrada específicos para se executado com êxito. Consulte a *Referência do criador de conteúdo* para obter mais informações.

Recursos

O Catalyst Process Automation Services oferece suporte aos recursos a seguir.

Mapeamento de modelos USM

O esquema de USM (Unified Service Model - Modelo de Serviço Unificado) modela os relacionamentos e as diretivas que controlam como o Catalyst Process Automation Services federa, correlaciona e reconcilia dados e recursos no CA Process Automation.

O Catalyst Process Automation Services oferece suporte aos seguintes tipos de mapeamento de modelos USM:

Tipo e propriedade de USM	Tipo e propriedade do CA Process Automation
ITActivityTemplate	Fluxograma
MdrProduct	"CA:00074"
MdrProdInstance	"CA:00074:01"
MdrElementID	ReferencePath + ReferenceName

UrlParams	URL separado UrlParams contém o URL separado do CA Process Automation. Cole o URL no navegador e efetue login no CA Process Automation. A interface de usuário exibe um objeto de automação.
CreationTimestamp	CreationDate
LastModTimestamp	ModifiedDate
LastModUserName	EditedBy
ActivityTypes	"Fluxo de trabalho"
DefinitionName	ReferencePath + ReferenceName
DefinitionVersion	Versão
ITActivity	Workflow
MdrProduct	"CA:00074"
MdrProdInstance	"CA:00074:01"
MdrElementID	Instância
UrlParams	URL separado UrlParams contém o URL separado do CA Process Automation. Cole o URL no navegador e efetue login no CA Process Automation. A interface de usuário exibe um objeto de automação.
CreationTimestamp	CreationDate
LastModTimestamp	ModifiedDate
LastModUserName	EditedBy
ActivityID	ROID
RuntimeName	Instância
RuntimeDiscriminator	ROID
ActivityTypes	"Fluxo de trabalho"
StateDescription	RunState
ActivityState:	RunState:
"Encerrado-Concluído"	"Concluído"
"Encerrado-Concluído"	"completedResponse"
"Encerrado-Abandonado"	"Anulado"

"Encerrado-Falhou"	"Falhou"
"Normal-Em execução"	"Em execução"
"Normal-Aguardando"	"Inativo"
"Obstruído"	"Suspenso"
"Obstruído"	"BreakPointSuspended"
"Normal-Aguardando"	"Bloqueado"
"AwaitingScheduling"	"Em fila"
BinaryRelationship	Workflow
MdrProduct	"CA:00074"
MdrProdInstance	"CA:00074:01"
MdrElementID	Instância (de fluxo de trabalho)
SourceMdrProduct	"CA:00074"
SourceMdrProdInstance	"CA:00074:01"
SourceMdrElementID	Instância (de fluxo de trabalho)
TargetMdrProduct	"CA:00074"
TargetMdrProdInstance	"CA:00074:01"
TargetMdrElementID	ReferencePath + ReferenceName (do fluxograma)
Semântica	"IsInstanceOf"
Alerta	Tarefa ou fluxo de trabalho
MdrProduct	"CA:00074"
MdrProdInstance	"CA:00074:01"
MdrElementID	Instância (fluxo de trabalho) ou "Tarefa": + TaskID (tarefa)
OccurrenceTimestamp	ModifiedDate (fluxo de trabalho) ou Iniciado (tarefa)
AlertType	"Risco" (fluxo de trabalho) ou "Informativo" (tarefa)
Gravidade	"Principal" (fluxo de trabalho) ou "Informativo" (tarefa)
AlertedMdrProduct	"CA:00074"
AlertedMdrProdInstance	"CA:00074:01"
AlertedMdrElementID	Instância

Resumo	Instância + RunState (fluxo de trabalho) ou TaskID + RunState + PromptPath (tarefa)
Mensagem	EventDescription (fluxo de trabalho) ou PromptTitle + PromptDescription (tarefa)
UrlParams	URL "separado" UrlParams contém o URL separado do CA Process Automation. Cole o URL no navegador e efetue login no CA Process Automation. A interface de usuário exibe um objeto de automação.

Recursos de consulta

O Catalyst Process Automation Services pode usar a operação get() nos tipos de USM suportados. As consultas oferecem suporte às propriedades "id" e "updatedAfter" do filtro básico. Um "get" para alertas retorna tarefas ou solicitações de interação pendentes.

Operações personalizadas

O Catalyst Process Automation Services oferece suporte a operações personalizadas. As operações personalizadas estão disponíveis para:

- Processos
- Solicitações iniciais
- Tarefas
- Importação/exportação de conteúdo
- Conjuntos de dados
- Configuração de módulos

Observação: as operações personalizadas exigem acesso seguro. Se o valor de "use.catalyst.claims.credentials" for verdadeiro, as requisições também serão necessárias para as credenciais do CA Process Automation.

No operador Execute (no módulo Catalyst), selecione uma operação personalizada na lista suspensa Operações.

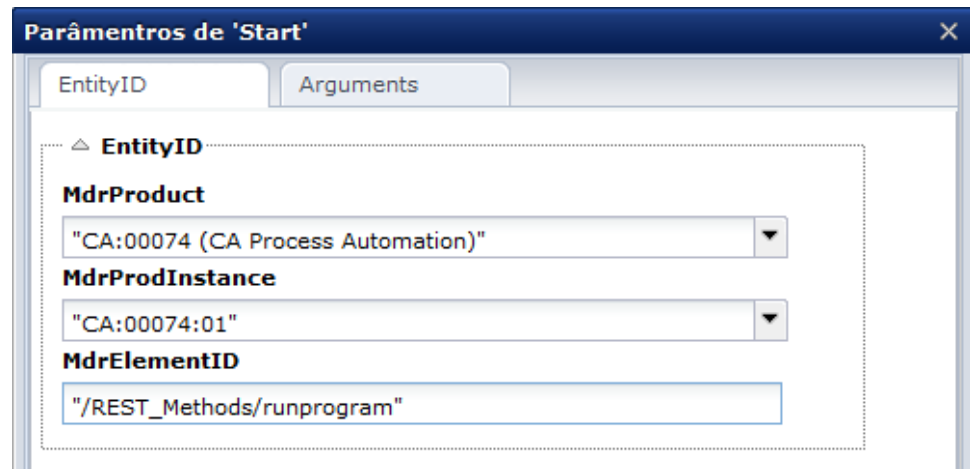
The image shows a screenshot of a web interface. It features a label "Operation:" followed by a dropdown menu. The dropdown menu is currently displaying the text "Export". The dropdown arrow is visible on the right side of the menu.

Operações do processo

As operações de processo permitem agir em um processo:

Iniciar um processo

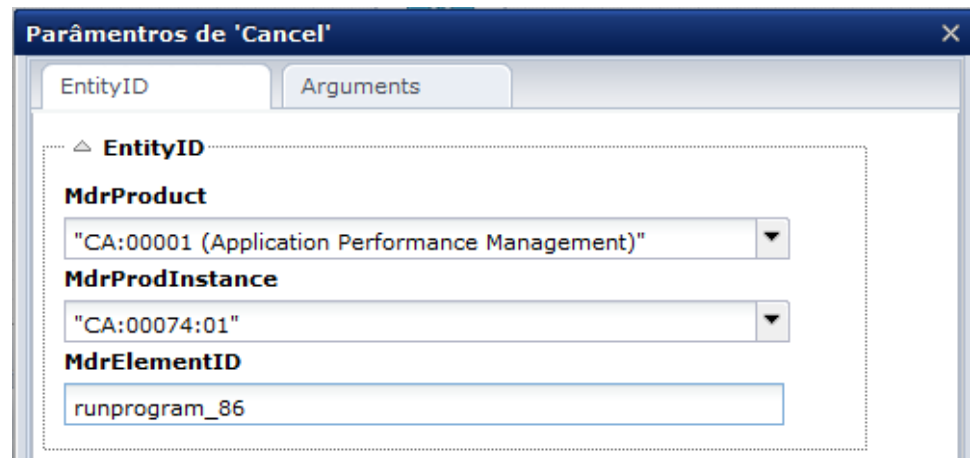
Usar o MdrElementID de um ITActivityTemplate.



The screenshot shows a dialog box titled "Parâmetros de 'Start'". It has two tabs: "EntityID" and "Arguments". The "EntityID" tab is active. Under the "EntityID" section, there are three fields: "MdrProduct" with the value "CA:00074 (CA Process Automation)", "MdrProdInstance" with the value "CA:00074:01", and "MdrElementID" with the value "/REST_Methods/runprogram".

Cancelar (abortar) um processo

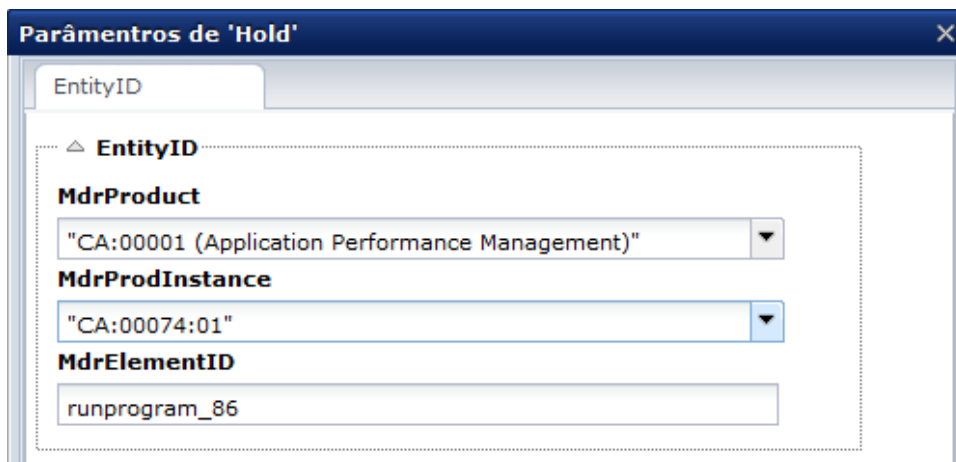
Usar o MdrElementID de um ITActivity.



The screenshot shows a dialog box titled "Parâmetros de 'Cancel'". It has two tabs: "EntityID" and "Arguments". The "EntityID" tab is active. Under the "EntityID" section, there are three fields: "MdrProduct" with the value "CA:00001 (Application Performance Management)", "MdrProdInstance" with the value "CA:00074:01", and "MdrElementID" with the value "runprogram_86".

Segurar (suspender) um processo

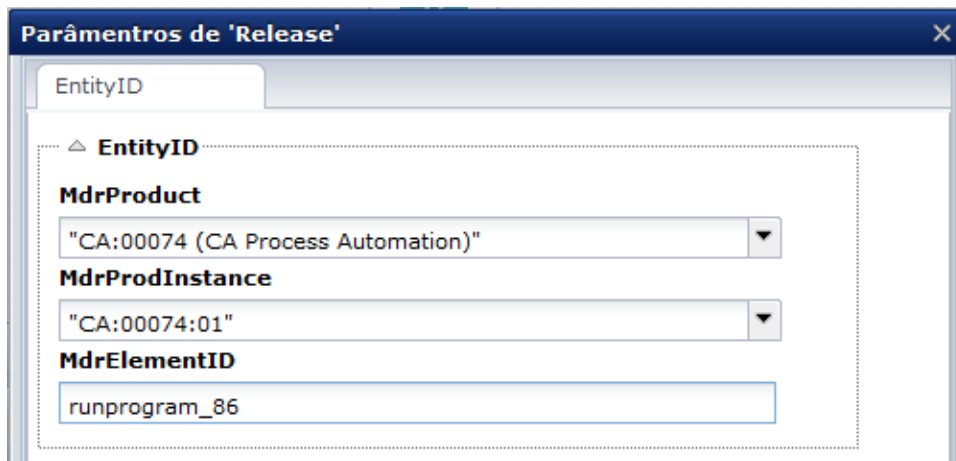
Usar o MdrElementID de um ITActivity.



The screenshot shows a dialog box titled "Parâmetros de 'Hold'". It has a tab labeled "EntityID". Below the tab, there is a tree view with "EntityID" expanded. Under "EntityID", there are three fields: "MdrProduct" with the value "CA:00001 (Application Performance Management)", "MdrProdInstance" with the value "CA:00074:01", and "MdrElementID" with the value "runprogram_86".

Liberar (retomar) um processo

Usar o MdrElementID de um ITActivity.



The screenshot shows a dialog box titled "Parâmetros de 'Release'". It has a tab labeled "EntityID". Below the tab, there is a tree view with "EntityID" expanded. Under "EntityID", there are three fields: "MdrProduct" with the value "CA:00074 (CA Process Automation)", "MdrProdInstance" with the value "CA:00074:01", and "MdrElementID" with the value "runprogram_86".

Operações de solicitação de início

As operações de solicitação inicial permitem executar as seguintes ações em formulários de solicitação inicial:

- QueryStartRequestForms
- SubmitStartRequestForm
- QueryStartRequests
- DequeueStartRequest
- AbortStartRequest

Exemplos

Os tópicos a seguir explicam como usar as operações de solicitação inicial com o operador de execução do Catalyst.

QueryStartRequestForms

Para consultar os formulários de solicitação inicial, use a operação QueryStartRequestForms.

Siga estas etapas:

1. Crie um processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.
4. Digite a segurança do Catalyst e os parâmetros de Executar necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha:

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

CatalystBrokerURL: `https://hostname:7443/ucf/BrokerService`

MdrProduct: `"CA:00074 (CA Process Automation)"`

MdrProdInstance: `"CA:00074:01"`

Operation Category: `"CA Process Automation"`

Operation: `QueryStartRequestForms`

5. Crie um filtro de recorrências que consulte formulários de solicitação inicial na biblioteca. O filtro pode restringir a pesquisa por caminho, de maneira recorrente, e por palavras-chave (tags).

- a. Nas propriedades de execução, clique em Parâmetros.

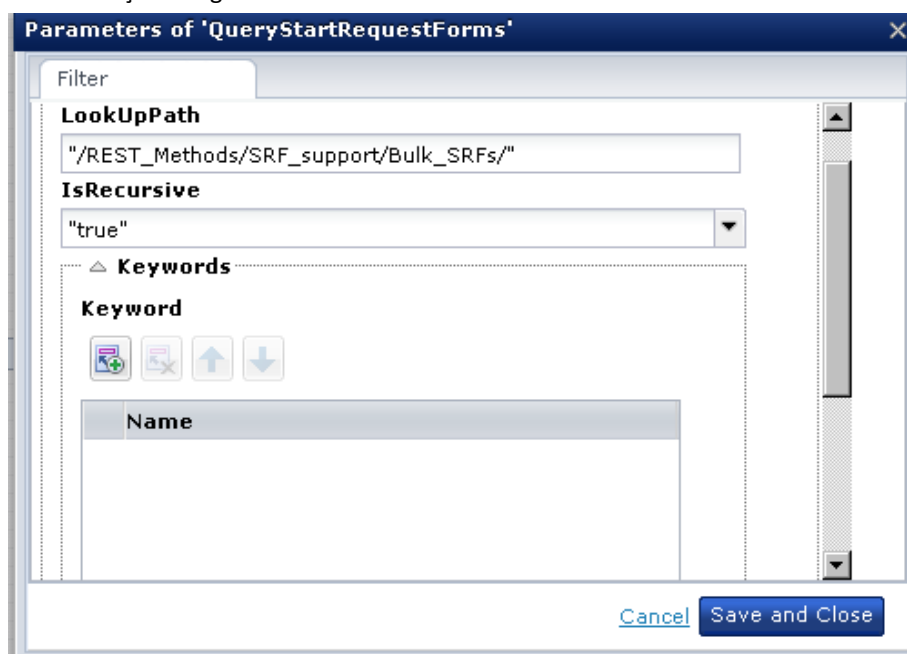
- b. Na guia Filtro, insira os parâmetros a seguir:

LookUpPath: define o caminho para consultar formulários de solicitação inicial. Neste exemplo, formulários de solicitação inicial estão sendo consultados na pasta pai "Bulk_SRFs".

IsRecursive: verdadeiro

Palavras-chave: define uma ou mais palavras-chave (como execução, na fila e concluído) com as quais consultar os formulários de solicitação inicial.

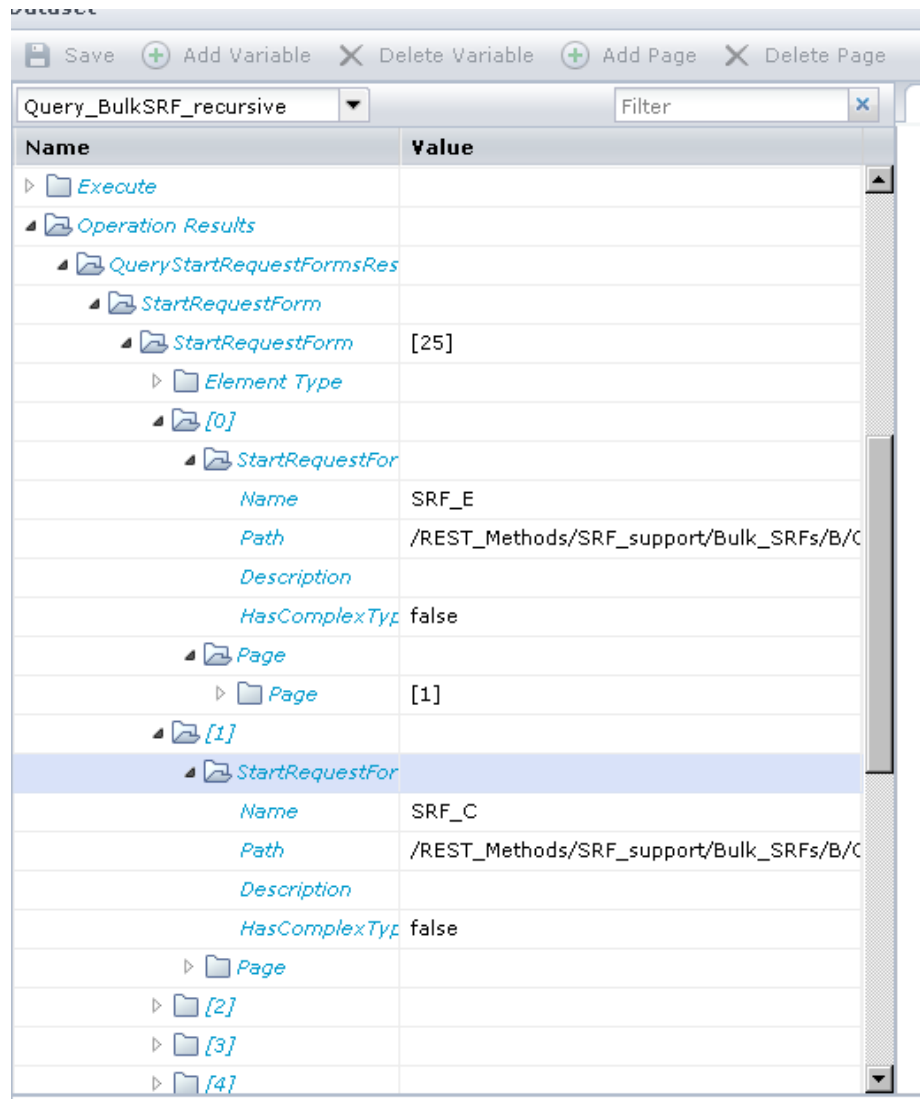
Você também pode realizar uma consulta sem palavras-chave, como na ilustração a seguir:



- c. Clique em Salvar e fechar.

6. Executar o processo.
7. Verifique a saída.

Por exemplo, na ilustração a seguir, existem 25 formulários de solicitação inicial (com variáveis) do LookUpPath fornecido:



SubmitStartRequestForm

Para enviar um formulário de solicitação inicial, use a operação SubmitStartRequestForm.

Siga estas etapas:

1. Crie um processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.
4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha:

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

CatalystBrokerURL: https://hostname:7443/ucf/BrokerService

MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

Operation Category: "CA Process Automation"

Operation: SubmitStartRequestForm

5. Envie um formulário de solicitação inicial para um estado em fila.
 - a. Nas propriedades de execução, clique em Parâmetros.
 - b. Na guia SubmitStartRequestFormRequest, insira os parâmetros a seguir:

Nome:

"SRF_Queue"

Caminho:

O caminho absoluto do formulário de solicitação inicial (você pode obter o caminho nas propriedades de um formulário de solicitação inicial). Por exemplo: SRF"/REST_Methods/SRF_support/SRFs_test/".

- c. Insira os argumentos na guia Argumentos. Por exemplo:

Nome:

Var_0

Valor:

from_proc

e aqui

Nome:

Var_1

Valor:

from_prc2

- d. Insira os parâmetros de data e hora na guia Opções. Por exemplo:

Data de início:

"2012-08-31" (aaaa-MM-dd)

Hora de início:

"12:25" (HH:mm)

Prioridade:

Sem valor

Observação: se você não definir a Data de início e a Hora de início, o CA Process Automation programará o formulário de solicitação inicial imediatamente.

6. Executar o processo.
7. Verifique a saída.

Quando o operador é executado, o CA Process Automation programa o formulário de solicitação inicial especificado no servidor. Conforme mostrado na ilustração a seguir, os parâmetros de saída incluem resultados como a ID de instâncias do formulário de solicitação inicial, o estado e detalhes do processo associado.

SubmitSRF_toQueuedState	
Name	Value
Execute	
Operation Results	
SubmitStartRequestFormResponse	
StartRequestStatus	
StartRequestStatus	
StartRequestStatus	
InteractionID	5948
Instance	SRF_Queue_5948
State	Queued
Earliest-start-time	2012-08-31T06:55:00.000Z
Start-time	
End-time	
RefProcess	/REST_Methods/SRF_support/SRFs_test/Proc_1

QueryStartRequests

A operação QueryStartRequests permite consultar uma instância de formulário de solicitação inicial em relação ao status da solicitação inicial e à ID da instância do processo associada.

Siga estas etapas:

1. Crie um processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.

4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha:

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

CatalystBrokerURL: `https://hostname:7443/ucf/BrokerService`

MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

Operation Category: "CA Process Automation"

Operation: QueryStartRequests

5. Identifique a instância do formulário de solicitação inicial que está no estado Em fila.
 - a. Nas propriedades de execução, clique em Parâmetros.
 - b. Na guia QueryStartRequestsRequest, insira os seguintes valores:

Instância: "SRF_Queue_x" (em que x é a ID da instância do formulário de solicitação inicial que você deseja consultar). Por exemplo, "SRF_Queue_5829".

IsArchived: "false"

6. Executar o processo.
7. Verifique a saída.

Na ilustração a seguir, os parâmetros de saída exibem o formulário de solicitação inicial no estado Em fila.

Query_SRFInstance	
Name	Value
StartRequestInstances	
StartRequestInstances	
StartRequestInstance	
StartRequestInstance	[1]
Element Type	
[0]	
StartRequestInstance	
Instance	SRF_Queue_5829
ProcessInstance	
State	Queued
Scheduled-time	2014-03-27T06:32:00.000Z
Start-time	
End-time	

Na ilustração a seguir, os parâmetros de saída exibem uma lista de todas as instâncias de formulário de solicitação inicial arquivadas. Para obter resultados arquivados, não inclua o nome da instância nos parâmetros de entrada, mas forneça o valor verdadeiro ou falso para o parâmetro IsArchived.

Dataset

Query_SRFInstance

Name	Value
StartRequestInstance	[30]
Element Type	
[0]	
StartRequestInst	
Instance	SRF_Run_361
ProcessInstance	Proc_Block_362
State	Running
Scheduled-time	2012-07-09T10:07:01.887Z
Start-time	2012-07-09T10:07:02.043Z
End-time	
[1]	
StartRequestInst	
Instance	SRF_Run_461
ProcessInstance	Proc_Block_462
State	Running
Scheduled-time	2012-07-10T02:17:02.333Z
Start-time	2012-07-10T02:17:02.617Z
End-time	
[2]	
[3]	
[4]	
[5]	
[6]	
[7]	
[8]	
[9]	
[10]	
[11]	

DequeueStartRequest

A operação DequeueStartRequest remove uma instância do formulário de solicitação inicial da fila.

Siga estas etapas:

1. Crie um processo.
2. Arraste o operador Executar para o Criador.

3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.
4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha:

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

CatalystBrokerURL: `https://hostname:7443/ucf/BrokerService`

MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

Operation Category: "CA Process Automation"

Operation: DequeueStartRequest

5. Retire da fila a instância do formulário de solicitação inicial que está no estado Em fila.
 - a. Nas propriedades de execução, clique em Parâmetros.
 - b. Na guia DequeueStartRequestRequest, insira o valor de instância "SRF_Queue_x" (em que x é a ID da instância do formulário de solicitação inicial a ser retirado da fila). Por exemplo, "SRF_Queue_5829."

6. Executar o processo.
7. Verifique a saída.

Na ilustração a seguir, os parâmetros de saída exibem o formulário de solicitação inicial retirado da fila. O Estado retornará como Com falha, pois a instância do formulário de solicitação inicial foi retirada da fila.

Observação: é possível retirar da fila apenas a instância do formulário de solicitação inicial que estiver na fila.

me	Value
<i>Execute</i>	
<i>Operation Results</i>	
<i>DequeueStartRequestRespon</i>	
<i>StartRequestInstance</i>	
<i>StartRequestInstance</i>	
<i>StartRequestInstanc</i>	
<i>Instance</i>	SRF_Queue_5829
<i>ProcessInstance</i>	
<i>State</i>	Failed
<i>Scheduled-time</i>	2014-03-27T06:32:00.000Z
<i>Start-time</i>	
<i>End-time</i>	

AbortStartRequest

Para encerrar uma instância do formulário de solicitação inicial, use a operação AbortStartRequest.

Siga estas etapas:

1. Crie um processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.

4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha:

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

CatalystBrokerURL: `https://hostname:7443/ucf/BrokerService`

MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

Operation Category: "CA Process Automation"

Operation: AbortStartRequest

5. Finalize a instância do formulário de solicitação inicial em execução:
 - a. Nas propriedades de execução, clique em Parâmetros.
 - b. Na guia AbortStartRequestRequest, insira o valor a seguir:

Instância:

"SRF_Run_x" (em que x é a ID da instância do formulário de solicitação inicial a ser retirado da fila). Por exemplo, "SRF_Run_3378."

6. Executar o processo.
7. Verifique a saída.

Na ilustração a seguir, os parâmetros de saída exibem o estado atual do formulário de solicitação inicial como Com falha, pois o estado Em execução foi encerrado.

Observação: você pode apenas encerrar a instância do formulário de solicitação inicial em execução.

Name	Value
> Execute	
▲ Operation Results	
▲ AbortStartRequestResponse	
▲ StartRequestInstance	
▲ StartRequestInstance	
▲ StartRequestInsta	
Instance	SRF_Run_3378
ProcessInstance	Proc_Block_3379
State	Failed
Scheduled-time	2012-07-30T12:40:32.083Z
Start-time	2012-07-30T12:40:32.850Z
End-time	

Operações de tarefas

As operações de tarefa permitem realizar as seguintes ações em tarefas:

QueryTasks

Consultar tarefas (solicitações de interação pendentes).

Parâmetros de 'QueryTasks'

QueryTasksReq...

TaskID

RootUUID

ProcessID

State

ReplyTask

Responder a uma tarefa com parâmetros e aprovação.

The screenshot shows a dialog box titled "Parâmetros de 'TakeTask'". It has a tab labeled "TakeTaskReque...". Below the tab, there is a label "TaskID" and a text input field containing the value "125".

TakeTask

Assumir uma tarefa.

The screenshot shows a dialog box titled "Parâmetros de 'TakeTask'". It has a tab labeled "TakeTaskReque...". Below the tab, there is a label "TaskID" and a text input field containing the value "125".

ReturnTask

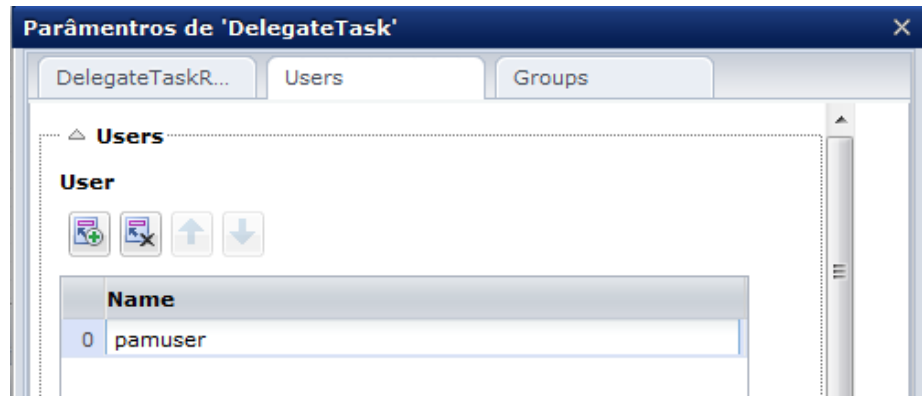
Retornar uma tarefa.

The screenshot shows a dialog box titled "Parâmetros de 'ReturnTask'". It has a tab labeled "ReturnTaskReq...". Below the tab, there is a label "TaskID" and a text input field containing the value "125".

DelegateTask

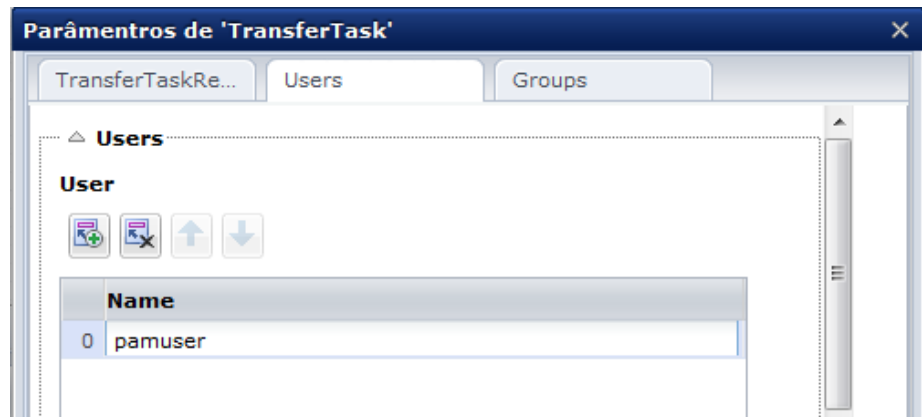
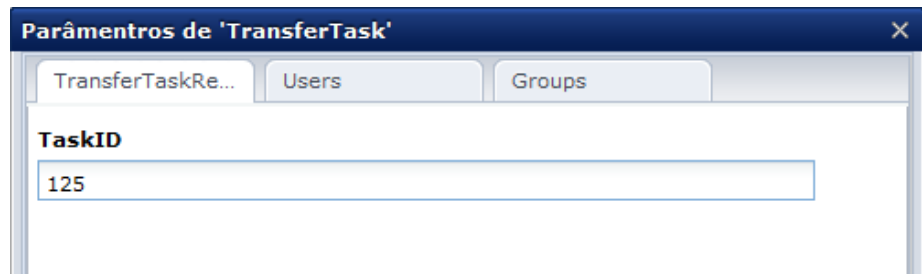
Delegar uma tarefa a um ou mais usuários ou grupos.

The screenshot shows a dialog box titled "Parâmetros de 'DelegateTask'". It has three tabs: "DelegateTaskR...", "Users", and "Groups". Below the tabs, there is a label "TaskID" and a text input field containing the value "125".



TransferTask

Transferir (isto é, atribuir) uma tarefa a um ou mais usuários ou grupos.



Operações de conteúdo

As operações de conteúdo permitem exportar o conteúdo de um ambiente do CA Process Automation e importá-lo em outro ambiente do CA Process Automation.

Importar

A operação Importar permite importar um arquivo local de objetos do CA Process Automation. Durante uma importação, você pode configurar <OverwriteAction> para uma das seguintes opções:

- Importar
- DoNotImport
- Importar e substituir

Usando o operador Executar, selecione a operação Importar e, em seguida, forneça os valores.

Exemplo

Os serviços RESTful permitem importar conteúdo de um local de servidor.

Siga estas etapas:

1. Crie um processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.
4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

afirma:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

CatalystBrokerURL: "https://hostname:7443/ucf/BrokerService"

MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

Operation Category: "CA Process Automation"

Operation: Import

5. Determine quais das opções a seguir devem ser utilizadas ao importar o conteúdo e, em seguida, preencha os campos de local de servidor:

- Importar de local do servidor
- Importar do URL HTTP

Local do servidor

- a. Nas propriedades de execução, clique em Parâmetros.
- b. Na guia ImportRequest, preencha os campos a seguir:

ImportLocation: define o local da biblioteca no qual importar o arquivo XML. Se a pasta especificada não existir, ela será criada pelo CA Process Automation. Por exemplo, "/test-import/".

SourceLocation: define a pasta de origem de uma das seguintes maneiras:

- Define o local do servidor a partir do qual importar o conteúdo. Por exemplo, "c:\export_import_test\export_test.xml".
- Define o URL HTTP para o local do qual importar o conteúdo. Por exemplo, "http://servername:8080/c2orepository/test/export_test.xml".

OverwriteAction: digite uma das opções a seguir para importar um arquivo de mesmo nome que existe no local especificado:

- **Importar:** aumenta a versão do objeto importado se existir um objeto de mesmo nome na pasta de importação especificada. Substitui a versão da release se ela já existir.
- **DoNotImport:** ignora a operação de importação se houver um objeto de mesmo nome na pasta de importação especificada.
- **ImportAndReplace:** substitui o objeto de mesmo nome na pasta de importação especificada. Exclui as versões anteriores do objeto.

SetCurrent: selecione "verdadeiro" para definir a versão importada como atual; caso contrário, selecione "falso".

MakeAvailable: selecione "verdadeiro" para disponibilizar o conteúdo na biblioteca. Selecione "falso" para tornar o conteúdo indisponível.

6. Executar o processo.

7. Verifique a saída.

A ilustração a seguir mostra uma mensagem de importação com êxito:

▸ Execute	
▾ Operation Results	
▾ ImportResponse	
▾ ImportResponse	
SuccessMessage	The specified object imported successfully.

Exportar

As operações de exportação permitem exportar um arquivo local de objetos do CA Process Automation para um arquivo local. O filtro pode usar o tipo de objeto para restringir os objetos a serem exportados.

Usando o operador Executar, selecione a operação Exportar e, em seguida, forneça os valores.

Exemplo

Para exportar conteúdo com um caminho relativo, use os serviços RESTful.

Siga estas etapas:

1. Crie um novo processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.
4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha:

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

CatalystBrokerURL: "https://hostname:7443/ucf/BrokerService"

MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

Operation Category: "CA Process Automation"

Operation: Export

5. Exporte o conteúdo com um caminho relativo.
 - a. Nas propriedades de execução, clique em Parâmetros.
 - b. Na guia ExportRequest, insira os seguintes valores:

FolderName: caminho absoluto da pasta ou do conteúdo. Por exemplo, "/GetOSInfo".

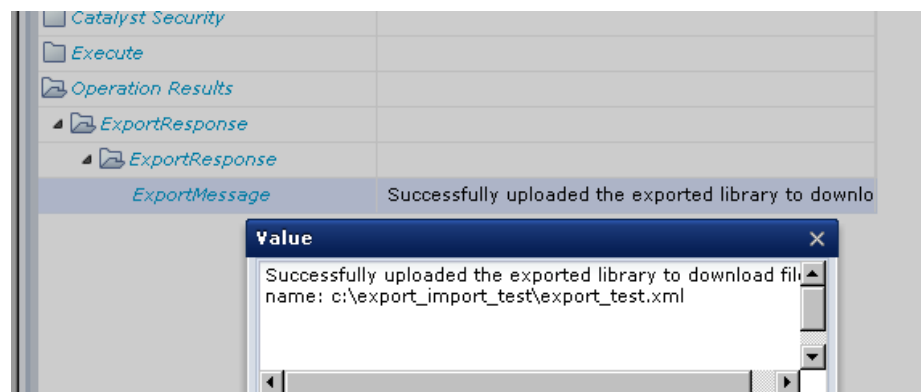
ExportAsContentPackage: exportar a pasta como pacote de conteúdo com um caminho relativo ou absoluto. Assume valor de verdadeiro ou falso. Por padrão, o valor é definido como falso.

IsAbsolute: selecione "falso" para exportar o conteúdo com um caminho relativo. Se você selecionar "verdadeiro", o conteúdo será importado com o caminho absoluto.
 - c. Na guia ExportFileName, insira os seguintes valores:

ExportLocation: local do servidor para o qual exportar o arquivo .xml. Por exemplo: "c:\export_import_test\export_test.xml".

OverwriteFile: seleciona se deseja substituir o arquivo existente no local determinado.
 - d. Para exportar todo o conteúdo a fim de incluir qualquer tipo de objeto, não forneça os tipos de objeto. Quando desejar exportar tipos de objetos específicos a partir de um determinado local, forneça os tipos, como Processo, Conjunto de dados.
6. Executar o processo.
7. Verifique a saída.

Na ilustração a seguir, uma mensagem de exportação com êxito é exibida com o local da exportação.



Operações do conjunto de dados

As operações do conjunto de dados permitem consultar e atualizar conjuntos de dados nomeados globais.

Exemplos

Os tópicos a seguir explicam como usar as operações do conjunto de dados com o operador Execute do Catalyst.

QueryDatasetNames

Use a operação QueryDatasetNames para pesquisar o caminho especificado para nomes do conjunto de dados.

Siga estas etapas::

1. Crie um novo processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.
4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

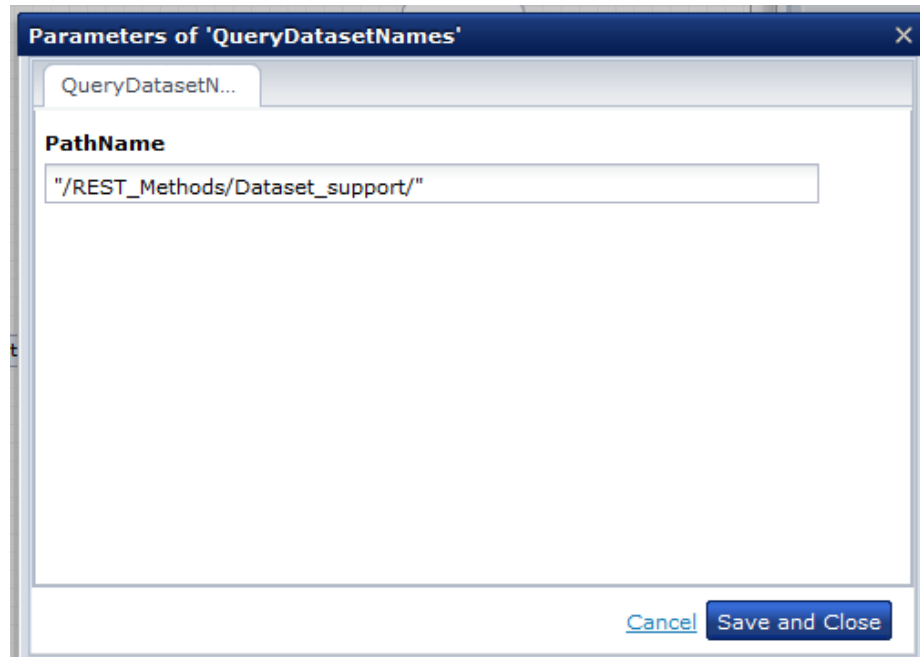
CatalystBrokerURL: https://hostname:7443/ucf/BrokerService

MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

Operation Category: "CA Process Automation"

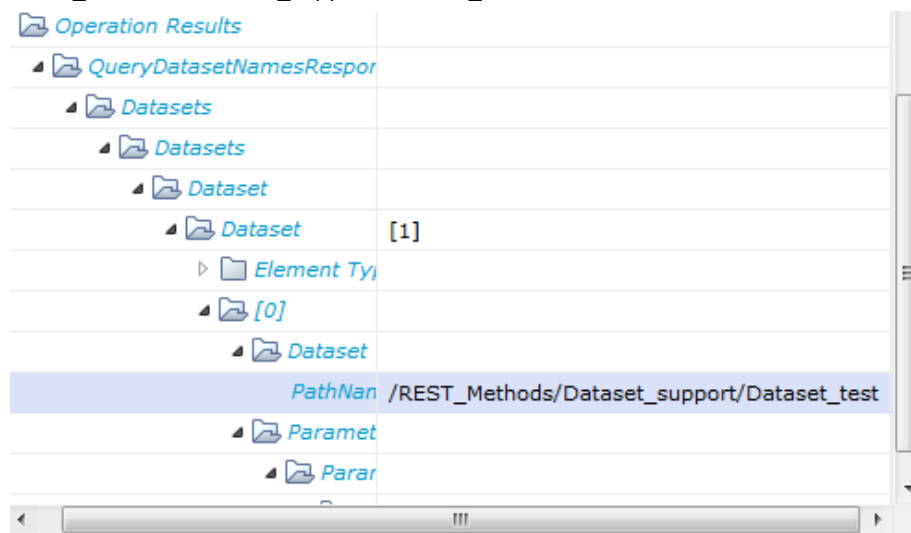
Operation: QueryDatasetNames



5. Executar o processo.
6. Verifique a saída.

Esta ilustração mostra o nome do conjunto de dados com o caminho absoluto no seguinte caminho Biblioteca:

/REST_Methods/Dataset_support/Dataset_test



QueryDatasetParameters

Use a operação QueryDatasetParameters para pesquisar o caminho especificado para parâmetros do conjunto de dados.

Siga estas etapas::

1. Crie um novo processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.
4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

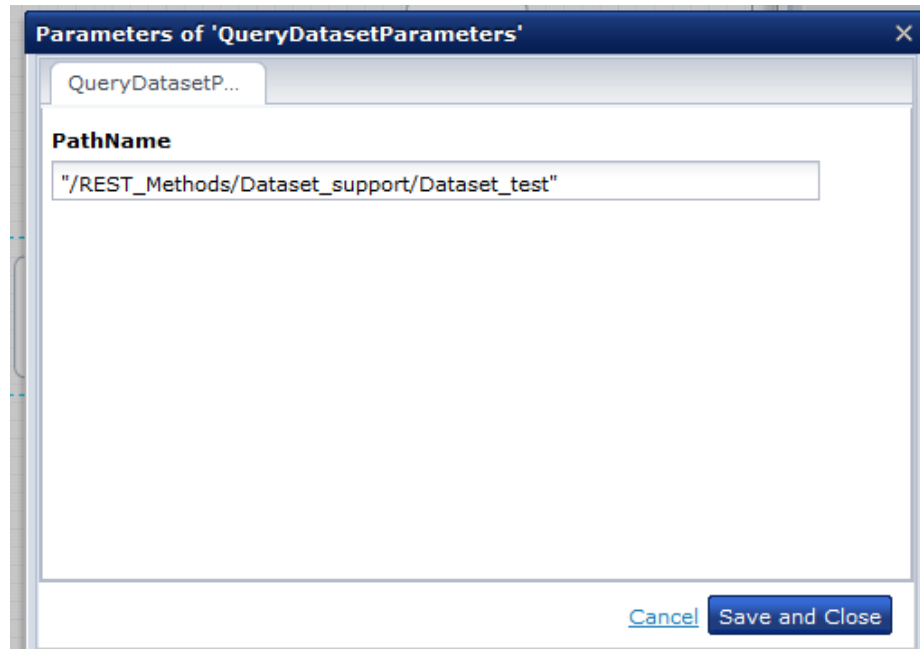
CatalystBrokerURL: https://hostname:7443/ucf/BrokerService

MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

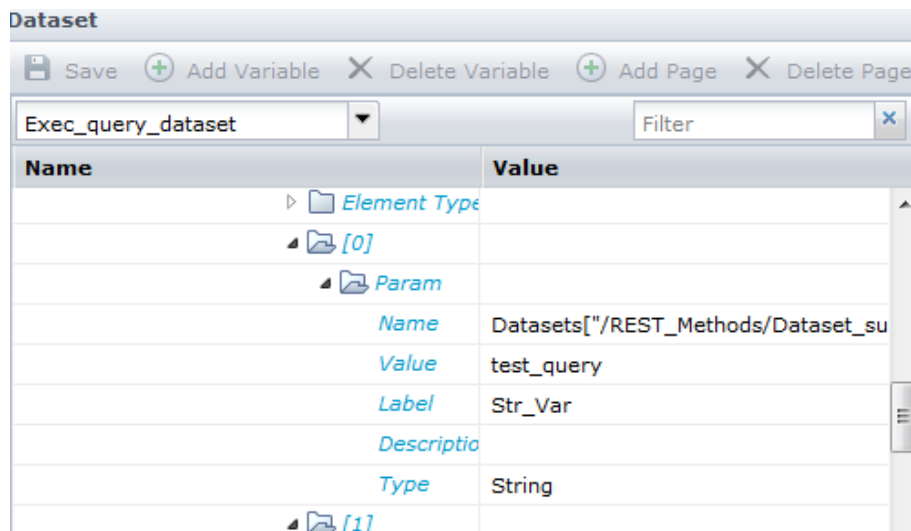
Operation Category: "CA Process Automation"

Operation: QueryDatasetParameters



5. Executar o processo.
6. Verifique a saída.

Esta ilustração mostra o nome do conjunto de dados com o Valor, Tipo:



UpdateDatasetParameters

Use a operação UpdateDatasetParameters para atualizar os parâmetros de um conjunto de dados.

Siga estas etapas::

1. Crie um novo processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.
4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

CatalystBrokerURL: https://hostname:7443/ucf/BrokerService

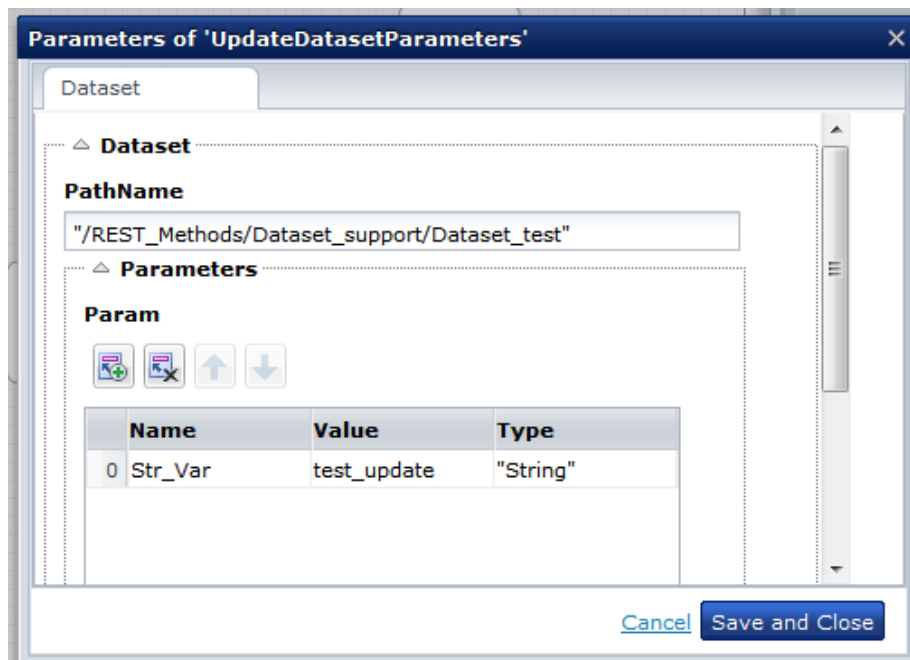
MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

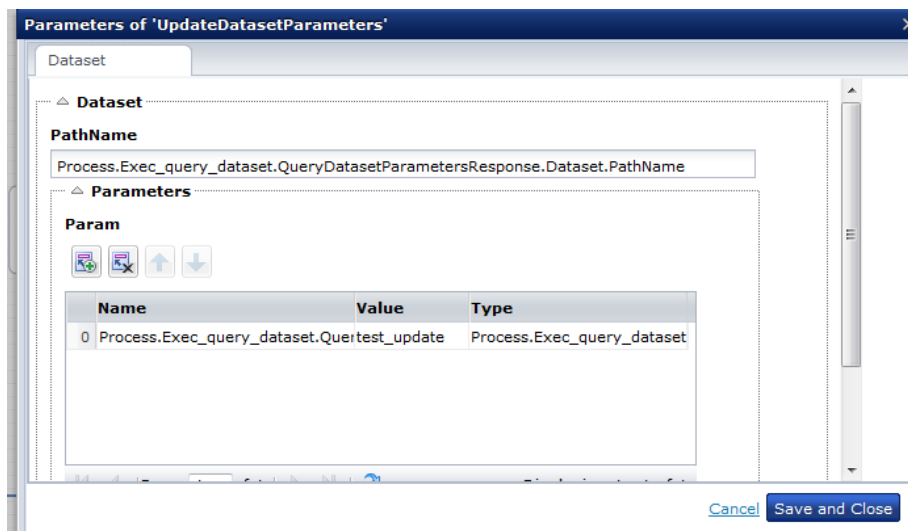
Operation Category: "CA Process Automation"

Operation: UpdateDatasetParameters

Este exemplo mostra o parâmetro do conjunto de dados do tipo "sequência de caracteres" com valor de sequência de caracteres:

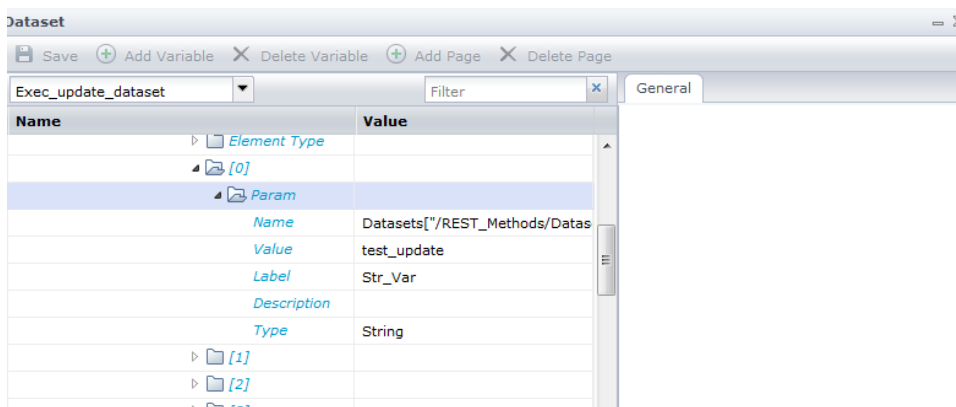


Este exemplo mostra como atualizar o parâmetro do conjunto de dados usando uma expressão:



5. Executar o processo.
6. Verifique a saída.

Esta ilustração mostra como o parâmetro do conjunto de dados do tipo "Sequência de caracteres" é atualizado com o valor fornecido "test_update".



QueryProcessDataset

Para retornar todos os parâmetros que um conjunto de dados contém, use a operação QueryDatasetParameters.

Siga estas etapas:

1. Crie um processo.
2. Arraste o operador Executar para o Criador.
3. Clique duas vezes no operador Executar para abrir a paleta Propriedades.
4. Insira os parâmetros necessários.

Segurança do Catalyst

Nome de usuário: *(nome de usuário do Catalyst)*

Senha: *(senha do Catalyst)*

Requisições:

Nome da requisição: Nome de usuário

Valor da requisição: pamadmin

Requisições de senha

Nome da requisição: Senha

Valor da requisição: pamadmin

Execute

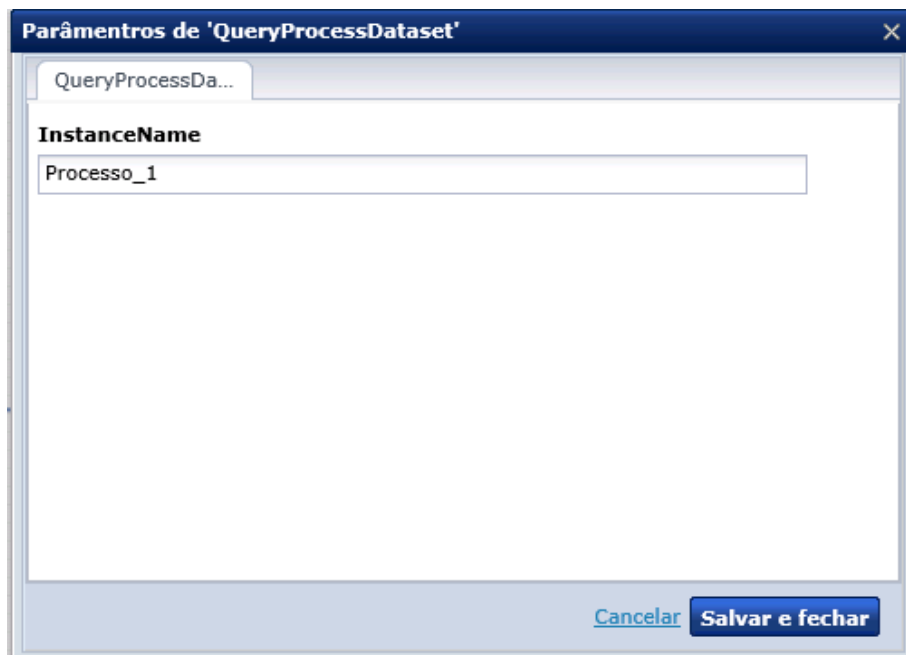
CatalystBrokerURL: https://hostname:7443/ucf/BrokerService

MdrProduct: "CA:00074 (CA Process Automation)"

MdrProdInstance: "CA:00074:01"

Operation Category: "CA Process Automation"

Operation: QueryProcessDataset



5. Executar o processo.
6. Verifique a saída.

Dataset

ExecuteQueryProcessDataset

Name	Value
System	
Catalyst Security	
Execute	
Operation Results	
QueryProcessDatasetResponse	
ProcessDataset	
ProcessDataset	
Parameters	
Parameters	
Param	
Param	[76]
Element Type	
[0]	
Param	
Name	Progress
Value	100.0
Label	Progress
Description	
Type	Double

Operações de configuração do módulo

As operações de configuração do módulo permitem executar ações em categorias (ou módulos) de operador:

QueryModuleConfigs

Consulta módulos.

(Opcional) O atributo <ModuleType> especifica se a configuração do módulo padrão ou do grupo de módulos personalizados é retornada. Por padrão, o método QueryModuleConfigs retorna todos os módulos disponíveis. O atributo <ModuleType> usa os seguintes valores:

Todas

Retorna a configuração de módulos padrão e personalizados.

Padrão

Retorna apenas a configuração de módulos padrão.

Personalizada

Retorna apenas a configuração de módulos personalizados.

QueryModuleConfigProperties

Pesquisa os módulos disponíveis e as propriedades de módulos personalizados.

UpdateModuleConfigProperties

Atualiza as propriedades dos módulos disponíveis e dos módulos personalizados.

Assinaturas de eventos

O Catalyst Process Automation Services oferece suporte às seguintes assinaturas de eventos:

- Evento "Criado" quando um ITActivity (processo) é iniciado.
- Evento "Modificado" quando um ITActivity (processo) é alterado (ou seja, quando ActivityState é alterado).
- Evento de alerta "Criado" quando um ITActivity (processo) falha ou é cancelado.

Observação: o evento "Excluído" não é aplicável.

Catalyst REST

Os serviços do Catalyst RESTful permitem acessar o Catalyst Process Automation Services. Para detectar os recursos da interface do Catalyst RESTful, inicie com o URL de base a partir de qualquer navegador e use as solicitações HTTP GET para percorrer os hiperlinks.

O URL de base da interface do Catalyst RESTful é:

```
http://<nome_do_host>:7000/node/rest/
```

O URL de base exibe os URLs dos módulos de recipiente do Catalyst. O recipiente do Catalyst hospeda módulos para o broker do Catalyst e o Catalyst Process Automation Services.

O status dos módulos pode ser exibido em:

```
http://<nome_do_host>:7000/node/rest/broker/Entity
```

O módulo do Catalyst Process Automation Services é "CA:00074_CA:00074:01":

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/
```

Os descritores do conector são acessados por meio da interface de metadados da API REST em:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01;metadata=descriptores
```

Por exemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<conndesc:descriptor
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:conndesc="http://www.ca.com/connex/conndesc"
  xsi:schemaLocation="http://www.ca.com/connex/conndesc
conn_desc.xsd"
  xmlns:usm="http://www.ca.com/usm"
  xmlns:xi="http://www.w3.org/2001/XInclude"
  xmlns:usm-core="http://ns.ca.com/2009/07/usm-core"
  xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"
  xmlns:pam="http://www.ca.com/pam">

  <ID>com.ca.c2o.mdr.UCFPAMConnector</ID>
  <version>4.0</version>
  <category>CA Process Automation</category>

  <configurationDesc xsi:type="usm:KeywordValuePairs">
  </configurationDesc>

  <connectorProperties>
    <property name = "BaseURI" value="http://pam-uri"/>
  </connectorProperties>

  <!-- ***** -->
  <!-- Process operations... -->
  <!-- ***** -->

  <!-- Start a Process -->
  <customOp ID="Start" name="Start" synchronous="true">
    <inputType xsi:type="pam-ops:StartRequest"/>
    <resultType xsi:type="pam-ops:StartResponse"/>
  </customOp>
  ...
```

O esquema das operações do conector é acessado por meio da interface de metadados da API REST em:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01;metadata=schema
```

Qualquer utilitário de esquema XML de terceiros pode usar o esquema para construir os parâmetros de entrada das operações.

Por exemplo:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://ns.ca.com/2011/09/pam-ops"
  xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"
  xmlns:usm-meta="http://ns.ca.com/2009/07/usm-metadata"

  xmlns:usm-meta2="http://ns.ca.com/2011/02/usm-metadata2"
  xmlns:usm-core="http://ns.ca.com/2009/07/usm-core" >

  <xs:import namespace="http://ns.ca.com/2009/07/usm-metadata"
  schemaLocation="usm-metadata-200907.xsd"/>
  <xs:import
  namespace="http://ns.ca.com/2011/02/usm-metadata2"
  schemaLocation="usm-metadata2-201102.xsd"/>
  <xs:import namespace="http://ns.ca.com/2009/07/usm-core"
  schemaLocation="usm-core-200907.xsd"/>

  <xs:complexType name="StartRequest">
    <xs:annotation>
      <xs:documentation>Request to start an entity, identified by its
  EntityID</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="EntityID" type="usm-core:EntityID">
        <xs:annotation>
          <xs:documentation>Identification of the entity which
  should be started (defined by the elements,
  MdrProduct-MdrProdInstance-MdrElementID).</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="Arguments" type="pam-ops:ArgumentList"
  minOccurs="0" maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>
  <xs:element name="StartRequest" type="pam-ops:StartRequest">
    <xs:annotation>
      <xs:documentation>Request to start an entity, identified by its
  EntityID</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:complexType name="StartResponse">
    <xs:annotation>
      <xs:documentation>If successfully started, a response
  containing the EntityID is returned</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="EntityID" type="usm-core:EntityID">
        <xs:annotation>
```

```
        <xs:documentation>Identification of the entity which was
requested to be started (defined by the elements,
MdrProduct-MdrProdInstance-MdrElementID).</xs:documentation>
    </xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
<xs:element name="StartResponse" type="pam-ops:StartResponse">
    <xs:annotation>
        <xs:documentation>If successfully started, a response
containing theEntityID is returned</xs:documentation>
    </xs:annotation>
</xs:element>
...
```

Iniciando com o URL da interface do Catalyst REST, é possível exibir o URL do Catalyst Process Automation Services. O URL do Catalyst Process Automation Services contém os URLs dos tipos e das operações suportados. Por exemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<module xmlns="http://ns.ca.com/2010/11/coreapi">
  <id>CA:00074_CA:00074:01</id>
  <mdrProduct>CA:00074</mdrProduct>
  <mdrProdInstance> CA:00074:01</mdrProdInstance>
  <type>
    <id>ITActivity</id>
    <name>ITActivity</name>
    <namespace>http://ns.ca.com/2009/07/usm-core</namespace>
    <prefix/>
    <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivity" rel="list"/>
    <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivity;metadata" rel="metadata"/>
    </type>
  <type>
    <id>ITActivityTemplate</id>
    <name>ITActivityTemplate</name>
    <namespace>http://ns.ca.com/2009/07/usm-core</namespace>
    <prefix/>
    <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivityTemplate" rel="list"/>
    <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivityTemplate;metadata" rel="metadata"/>
    </type>
  <operation>
    <id>Release</id>
    <name>Release</name>
    <synchronous>>true</synchronous>
    <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/_ops
/Release" rel="self"/>
    </operation>
```

Exibir processos

Para exibir os processos do CA Process Automation em um navegador, use o seguinte URL:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/ITActivityTe
mplate
```

Por exemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<enumeration xmlns="http://ns.ca.com/2010/11/coreapi">
  <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivityTemplate?count=-1&start=0" rel="all"/>
  <usm-core:ITActivityTemplate
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core">
    <usm-core:MdrProduct>CA:00074</usm-core:MdrProduct>
    <usm-core:MdrProdInstance>
CA:00074:01</usm-core:MdrProdInstance>
    <usm-core:MdrElementID>/Test/TestExec1</usm-core:MdrElementID>
    <usm-core:Label>Workflow:/Test/TestExec1</usm-core:Label>

<usm-core:CreationTimestamp>2011-11-28T13:07:03</usm-core:Creation
Timestamp>

<usm-core:LastModTimestamp>2011-11-28T13:07:19</usm-core:LastModTi
mestamp>
  <usm-core:InstanceName>Workflow:1</usm-core:InstanceName>

<usm-core:DefinitionName>/Test/TestExec1</usm-core:DefinitionName>
  <usm-core:DefinitionVersion>1</usm-core:DefinitionVersion>
  <usm-core:ActivityTypes>Workflow</usm-core:ActivityTypes>
  <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivityTemplate/Test/TestExec1" rel="self"/>
  </usm-core:ITActivityTemplate>
  <usm-core:ITActivityTemplate
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core">
    <usm-core:MdrProduct>CA:00074</usm-core:MdrProduct>
    <usm-core:MdrProdInstance>
CA:00074:01</usm-core:MdrProdInstance>

<usm-core:MdrElementID>/Test/TestGetITActivity</usm-core:MdrElemen
tID>

<usm-core:Label>Workflow:/Test/TestGetITActivity:1</usm-core:Label
>

<usm-core:CreationTimestamp>2011-11-29T20:47:50</usm-core:Creation
Timestamp>

<usm-core:LastModTimestamp>2011-11-29T20:48:02</usm-core:LastModTi
mestamp>
  <usm-core:InstanceName>Workflow:1</usm-core:InstanceName>

<usm-core:DefinitionName>/Test/TestGetITActivity</usm-core:Definit
ionName>
```

```

    <usm-core:DefinitionVersion>1</usm-core:DefinitionVersion>
    <usm-core:ActivityTypes>Workflow</usm-core:ActivityTypes>
    <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivityTemplate/Test/TestGetITActivity" rel="self"/>
    </usm-core:ITActivityTemplate>
    <usm-core:ITActivityTemplate
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core">
    <usm-core:MdrProduct>CA:00074</usm-core:MdrProduct>

<usm-core:MdrProdInstance>CA:00074:01</usm-core:MdrProdInstance>Mdr
rProdInstance>CA:00074:01</usm-core:MdrProdInstance>

<usm-core:MdrElementID>/Test/TestExecRelease</usm-core:MdrElementI
D>

<usm-core:Label>Workflow:/Test/TestExecRelease:1</usm-core:Label>

<usm-core:CreationTimestamp>2011-12-01T15:10:47</usm-core:Creation
Timestamp>

<usm-core>LastModTimestamp>2011-12-01T15:11:03</usm-core>LastModTi
mestamp>
    <usm-core:InstanceName>Workflow:1</usm-core:InstanceName>

<usm-core:DefinitionName>/Test/TestExecRelease</usm-core:Definitio
nName>
    <usm-core:DefinitionVersion>1</usm-core:DefinitionVersion>
    <usm-core:ActivityTypes>Workflow</usm-core:ActivityTypes>
    <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivityTemplate/Test/TestExecRelease" rel="self"/>
    </usm-core:ITActivityTemplate>
</enumeration>

```

Exibir instâncias de processos

Para exibir as instâncias de processos do CA Process Automation em um navegador, use o seguinte URL:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/ITActivity
```

Por exemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<enumeration xmlns="http://ns.ca.com/2010/11/coreapi">
  <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivity?count=-1&start=0" rel="all"/>
  <usm-core:ITActivity
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core">
    <usm-core:MdrProduct>CA:00074</usm-core:MdrProduct>

    <usm-core:MdrProdInstance>CA:00074:01</usm-core:MdrProdInstance>Mdr
ProdInstance>CA:00074:01</usm-core:MdrProdInstance>
    <usm-core:MdrElementID>TestGetPAM_140</usm-core:MdrElementID>

    <usm-core:Label>Workflow:TestGetPAM_140:140:140</usm-core:Label>

    <usm-core:CreationTimestamp>2011-12-01T12:43:12</usm-core:Creation
Timestamp>

    <usm-core:LastModTimestamp>2011-12-01T12:44:01</usm-core:LastModTi
mestamp>
    <usm-core:LastModUserName>pamadmin</usm-core:LastModUserName>

    <usm-core:InstanceName>TestGetPAM_140:140:Workflow</usm-core:Insta
nceName>
    <usm-core:ActivityID>140</usm-core:ActivityID>
    <usm-core:RuntimeName>TestGetPAM_140</usm-core:RuntimeName>

    <usm-core:RuntimeDiscriminator>140</usm-core:RuntimeDiscriminator>
    <usm-core:ActivityTypes>Workflow</usm-core:ActivityTypes>

    <usm-core:ActivityState>Finished-Completed</usm-core:ActivityState
>

    <usm-core:StateDescription>Completed</usm-core:StateDescription>
    <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivity/TestGetPAM_140" rel="self"/>
    </usm-core:ITActivity>
  </usm-core:ITActivity
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core">
    <usm-core:MdrProduct>CA:00074</usm-core:MdrProduct>

    <usm-core:MdrProdInstance>CA:00074:01</usm-core:MdrProdInstance>Mdr
ProdInstance>CA:00074:01</usm-core:MdrProdInstance>
    <usm-core:MdrElementID>TestGetPAM_135</usm-core:MdrElementID>

    <usm-core:Label>Workflow:TestGetPAM_135:135:135</usm-core:Label>
```

```
<usm-core:CreationTimestamp>2011-12-01T11:59:53</usm-core:CreationTimestamp>

<usm-core>LastModTimestamp>2011-12-01T12:01:11</usm-core>LastModTimestamp>
  <usm-core>LastModUserName>pamadmin</usm-core>LastModUserName>

<usm-core:InstanceName>TestGetPAM_135:135:Workflow</usm-core:InstanceName>
  <usm-core:ActivityID>135</usm-core:ActivityID>
  <usm-core:RuntimeName>TestGetPAM_135</usm-core:RuntimeName>

<usm-core:RuntimeDiscriminator>135</usm-core:RuntimeDiscriminator>
  <usm-core:ActivityTypes>Workflow</usm-core:ActivityTypes>

<usm-core:ActivityState>Finished-Failed</usm-core:ActivityState>
  <usm-core:StateDescription>Failed</usm-core:StateDescription>
  <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITActivity/TestGetPAM_135" rel="self"/>
  </usm-core:ITActivity>
  <usm-core:ITActivity
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core">
  <usm-core:MdrProduct>CA:00074</usm-core:MdrProduct>

<usm-core:MdrProdInstance>CA:00074:01</usm-core:MdrProdInstance>MdrProdInstance>CA:00074:01</usm-core:MdrProdInstance>

<usm-core:MdrElementID>TestGetITActivity_130</usm-core:MdrElementID>

<usm-core:Label>Workflow:TestGetITActivity_130:130:130</usm-core:Label>

<usm-core:CreationTimestamp>2011-12-01T11:53:44</usm-core:CreationTimestamp>

<usm-core>LastModTimestamp>2011-12-01T11:59:14</usm-core>LastModTimestamp>
  <usm-core>LastModUserName>pamadmin</usm-core>LastModUserName>

<usm-core:InstanceName>TestGetITActivity_130:130:Workflow</usm-core:InstanceName>
  <usm-core:ActivityID>130</usm-core:ActivityID>

<usm-core:RuntimeName>TestGetITActivity_130</usm-core:RuntimeName>

<usm-core:RuntimeDiscriminator>130</usm-core:RuntimeDiscriminator>
```

```
<usm-core:ActivityTypes>Workflow</usm-core:ActivityTypes>
<usm-core:ActivityState>Finished-Completed</usm-core:ActivityState
>
<usm-core:StateDescription>Completed</usm-core:StateDescription>
  <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITAc
tivity/TestGetITActivity_130" rel="self"/>
  </usm-core:ITActivity>
</enumeration>
```

Exibir relacionamentos de processos

Para exibir os relacionamentos de processos do CA Process Automation em um navegador, use o seguinte URL:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/BinaryRelati
onship
```

Por exemplo:

```
<?xml version="1.0" encoding="utf-8"?>
<enumeration xmlns="http://ns.ca.com/2010/11/coreapi">
  <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:0074_CA:0074:
01/BinaryRelationship?count=-1&start=0" rel="all"/>
  <usm-core:BinaryRelationship
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core">
    <usm-core:MdrProduct>CA:00074</usm-core:MdrProduct>

<usm-core:MdrProdInstance>CA:00074:01</usm-core:MdrProdInstance>

<usm-core:MdrElementID>TestGetITActivityTemplate_1</usm-core:MdrEl
ementID>
  <usm-core:Label>IsInstanceOf</usm-core:Label>
  <usm-core:InstanceName/>

<usm-core:SourceMdrProduct>CA:00074</usm-core:SourceMdrProduct>

<usm-core:SourceMdrProdInstance>CA:0074_CA:0074:01</usm-core:Sourc
eMdrProdInstance>

<usm-core:SourceMdrElementID>TestGetITActivityTemplate_1</usm-core
:SourceMdrElementID>

<usm-core:TargetMdrProduct>CA:00074</usm-core:TargetMdrProduct>

<usm-core:TargetMdrProdInstance>CA:0074_CA:0074:01</usm-core:Targe
tMdrProdInstance>

<usm-core:TargetMdrElementID>/Tests/TestGetITActivityTemplate</usm
-core:TargetMdrElementID>
  <usm-core:Semantic>IsInstanceOf</usm-core:Semantic>
  <link
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:0074_CA:0074:
01/BinaryRelationship/TestGetITActivityTemplate_1" rel="self"/>
  </usm-core:BinaryRelationship>
  <usm-core:BinaryRelationship
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core">
    <usm-core:MdrProduct>CA:00074</usm-core:MdrProduct>

<usm-core:MdrProdInstance>CA:00074:01</usm-core:MdrProdInstance>

<usm-core:MdrElementID>TestGetITActivity_1</usm-core:MdrElementID>
  <usm-core:Label>IsInstanceOf</usm-core:Label>
  <usm-core:InstanceName/>

<usm-core:SourceMdrProduct>CA:00074</usm-core:SourceMdrProduct>
```

```
<usm-core:SourceMdrProdInstance>CA:0074_CA:0074:01</usm-core:SourceMdrProdInstance>

<usm-core:SourceMdrElementID>TestGetITActivity_1</usm-core:SourceMdrElementID>

<usm-core:TargetMdrProduct>CA:0074</usm-core:TargetMdrProduct>

<usm-core:TargetMdrProdInstance>CA:0074_CA:0074:01</usm-core:TargetMdrProdInstance>

<usm-core:TargetMdrElementID>/Tests/TestGetITActivity</usm-core:TargetMdrElementID>
  <usm-core:Semantic>IsInstanceOf</usm-core:Semantic>
  <link
href="http://mulwi01-w500:7000/node/rest/CA:0074_CA:0074_CA:0074:01/BinaryRelationship/TestGetITActivity_1" rel="self"/>
  </usm-core:BinaryRelationship>
</enumeration>
```

Exibir um objeto específico

É possível exibir qualquer objeto específico em um navegador usando estes formatos de URL:

```
http://<hostname>:7000/node/rest/CA:0074_CA:0074:01/<type>/<id>
```

```
http://<hostname>:7000/node/rest/CA:0074_CA:0074:01/<type>?id=<id>
```

em que:

- `<type>` é o tipo de USM
- `<id>` é o valor MdrElementID

Por exemplo, use o seguinte URL para exibir uma instância do processo (ITActivity) com a ID "TestSubscribeAlert_59":

```
http://<hostname>:7000/node/rest/CA:0074_CA:0074:01/ITActivity/TestSubscribeAlert_59
```

```
http://<hostname>:7000/node/rest/CA:0074_CA:0074:01/ITActivity?id=TestSubscribeAlert_59
```

Ao atualizar o navegador, é possível monitorar o status da instância do processo (nas propriedades ActivityState ou StateDescription da ITActivity).

```
<?xml version="1.0" encoding="utf-8"?>
<usm-core:ITActivity
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core">
  <usm-core:MdrProduct>CA:00074</usm-core:MdrProduct>
  <usm-core:MdrProdInstance>CA:00074:01</usm-core:MdrProdInstance>

  <usm-core:MdrElementID>TestSubscribeAlert_59</usm-core:MdrElementID>

  <usm-core:Label>Workflow:TestSubscribeAlert_59:59:59</usm-core:Label>

  <usm-core:CreationTimestamp>2012-04-18T15:57:38</usm-core:CreationTimestamp>

  <usm-core>LastModTimestamp>2012-04-18T15:59:18</usm-core>LastModTimestamp>
  <usm-core>LastModUserName>pamadmin</usm-core>LastModUserName>

  <usm-core:InstanceName>TestSubscribeAlert_59:59:Workflow</usm-core:InstanceName>
  <usm-core:ActivityID>59</usm-core:ActivityID>

  <usm-core:RuntimeName>TestSubscribeAlert_59</usm-core:RuntimeName>

  <usm-core:RuntimeDiscriminator>59</usm-core:RuntimeDiscriminator>
  <usm-core:ActivityTypes>Workflow</usm-core:ActivityTypes>

  <usm-core:ActivityState>Finished-Completed</usm-core:ActivityState>
  <usm-core:StateDescription>Completed</usm-core:StateDescription>
  <link xmlns="http://ns.ca.com/2010/11/coreapi"
href="http://mulwi01-w500:7000/node/rest/CA:00074_CA:00074:01/ITActivity/TestSubscribeAlert_59" rel="self"/>
</usm-core:ITActivity>
```

Executar operações de conector

Para executar operações de conector, envie solicitações HTTP POST usando o URL da operação.

Observação: apenas as operações execute requerem solicitações POST. Todas as outras solicitações usam GET.

Por exemplo, para iniciar um processo, envie uma solicitação HTTP POST ao seguinte URL:

`http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/Start`

Incluir um cabeçalho de solicitação:

`Content-Type=application/xml`

Incluir um corpo de solicitação que contenha uma operação StartRequest que inclui os seguintes itens:

- O nome do processo no elemento MdrElementID
- Os parâmetros no elemento Arguments

Por exemplo:

```
<StartRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="p1:StartRequest"
xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <EntityID>

<MdrElementID>/Tests/TestGetITActivityTemplate</MdrElementID>
  </EntityID>
  <Arguments>
    <Argument>
      <Name>arg1</Name>
      <Value>val1</Value>
    </Argument>
    <Argument>
      <Name>arg2</Name>
      <Value>val2</Value>
    </Argument>
    <Argument>
      <Name>arg3</Name>
      <Value>val3</Value>
    </Argument>
  </Arguments>
</StartRequest>
```

Iniciar processo

É possível usar qualquer ferramenta ou aplicativo do cliente HTTP para enviar a solicitação de iniciar o processo.

Por exemplo, para iniciar um processo, envie uma solicitação HTTP POST ao seguinte URL:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/Start
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

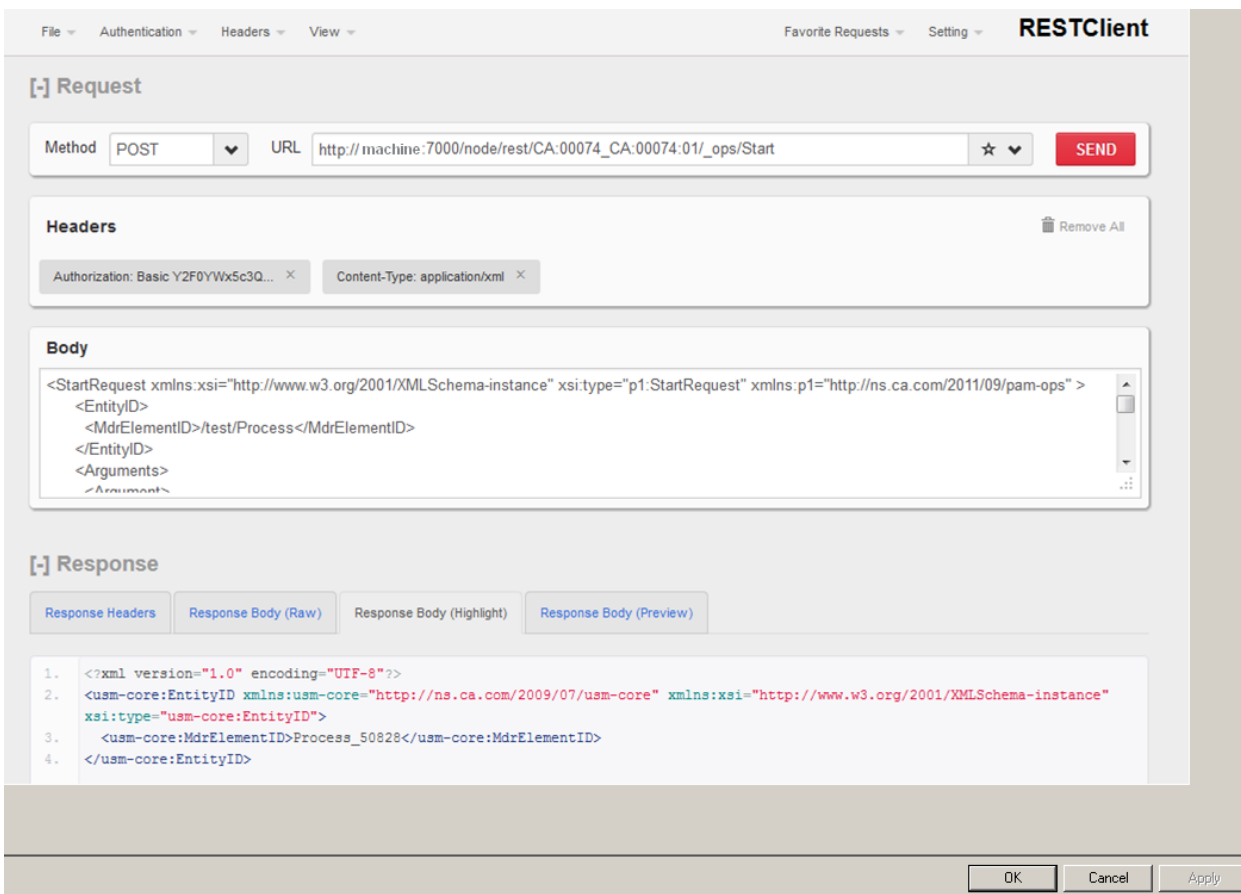
Por exemplo:

```
<StartRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="p1:StartRequest"
xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <EntityID>
    <MdrElementID>/test/Process</MdrElementID>
  </EntityID>
  <Arguments>
    <Argument>
      <Name>arg1</Name>
      <Value>val1</Value>
    </Argument>
    <Argument>
      <Name>arg2</Name>
      <Value>val2</Value>
    </Argument>
    <Argument>
      <Name>arg3</Name>
      <Value>val3</Value>
    </Argument>
  </Arguments>
</StartRequest>
```

A solicitação retorna o seguinte XML:

```
<usm-core:EntityID xsi:type="usm-core:EntityID">
<usm-core:MdrElementID>Process_51268</usm-core:MdrElementID>
</usm-core:EntityID>
```

O Firefox REST Client é mostrado neste exemplo:



Em espera

Para manter (ou suspender) um processo, especifique a ID do processo na propriedade MdrElementID.

Por exemplo, para manter um processo, envie uma solicitação HTTP POST para o seguinte URL:

http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/Hold

Incluir um cabeçalho de solicitação:

Content-Type=application/xml

Por exemplo:

```
<HoldRequest xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xsi:type="pl:HoldRequest"
             xmlns:pl="http://ns.ca.com/2011/09/pam-ops" >
  <EntityID>
    <MdrProduct/>
    <MdrProdInstance/>
    <MdrElementID>TestIRFPrompt_140</MdrElementID>
  </EntityID>
</HoldRequest>
```

A solicitação retorna o seguinte XML:

```
<usm-core:EntityID
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
             xsi:type="usm-core:EntityID">
  <usm-core:MdrProduct/>
  <usm-core:MdrProdInstance/>
  <usm-core:MdrElementID>TestIRFPrompt_140</usm-core:MdrElementID>
</usm-core:EntityID>
```

Release

É possível liberar (ou retomar) um processo, especificando a ID do processo na propriedade MdrElementID.

Por exemplo, para liberar um processo, envie uma solicitação HTTP POST ao seguinte URL:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/Release
```

Incluir um cabeçalho de solicitação:

Content-Type=application/xml

Por exemplo:

```
<ReleaseRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:ReleaseRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <EntityID>
    <MdrProduct/>
    <MdrProdInstance/>
    <MdrElementID>TestIRFPrompt_140</MdrElementID>
  </EntityID>
</ReleaseRequest>
```

A solicitação retorna o seguinte XML:

```
<usm-core:EntityID
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="usm-core:EntityID">
  <usm-core:MdrProduct/>
  <usm-core:MdrProdInstance/>
  <usm-core:MdrElementID>TestIRFPrompt_140</usm-core:MdrElementID>
</usm-core:EntityID>
```

Cancelar

Para cancelar (ou interromper) um processo, especifique a ID do processo na propriedade MdrElementID.

Por exemplo, para cancelar um processo, envie uma solicitação HTTP POST ao seguinte URL:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/Cancel
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<CancelRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:CancelRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <EntityID>
    <MdrProduct/>
    <MdrProdInstance/>
    <MdrElementID>TestIRFPrompt_140</MdrElementID>
  </EntityID>
</CancelRequest>
```

A solicitação retorna o seguinte XML:

```
<usm-core:EntityID
xmlns:usm-core="http://ns.ca.com/2009/07/usm-core"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="usm-core:EntityID">
  <usm-core:MdrProduct/>
  <usm-core:MdrProdInstance/>
  <usm-core:MdrElementID>TestIRFPrompt_140</usm-core:MdrElementID>
</usm-core:EntityID>
```

Iniciar solicitação de exemplos do REST

Os exemplos a seguir demonstram como usar serviços RESTful com formulários de solicitação inicial.

QueryStartRequestForms

É possível usar um filtro opcional com um caminho de biblioteca para consultar os formulários de solicitação inicial.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/QueryStartRequestForms
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<QueryStartRequestFormsRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="p1:QueryStartRequestFormsRequest"
xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <Filter>
    <LookUpPath>/TestSRF</LookUpPath>
    <IsRecursive>>true</IsRecursive>
    [set the value for HTML meta tag keywords]
    <Keyword>
      <Name>TagValue</Name>
    </Keyword>
  </Keywords>
</Filter>
</QueryStartRequestFormsRequest>
```

Conforme exibido no exemplo a seguir, a resposta de QueryStartRequestForms inclui os nomes e os tipos do parâmetro do formulário de solicitação inicial (mostrados em **negrito** no exemplo). A operação SubmitStartRequestForm usa os nomes e os tipos do parâmetro.

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:QueryStartRequestFormsResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="pam-ops:QueryStartRequestFormsResponse">
  <StartRequestForms>
    <StartRequestForm>
      <Name>SRF</Name>
      <Path>/TestSRF</Path>
      <Description>my test srf description</Description>
      <Pages>
        <HasComplexType>>false</HasComplexType>
        <Page>
          <Name>TestPage</Name>
          <Param>
            <Name>Var_0</Name>
            <Label>Text Field</Label>
            <Description/>
            <IsReadOnly>>false</IsReadOnly>
            <Type>String</Type>
          </Param>
          <Param>
            <Name>Var_1</Name>
            <Label>Check Box</Label>
            <Description/>
            <IsReadOnly>>false</IsReadOnly>
            <Type>Boolean</Type>
          </Param>
          <Param>
            <Name>Var_2</Name>
            <Label>Select</Label>
            <Description/>
            <IsReadOnly>>false</IsReadOnly>
            <Type>String</Type>
          </Param>
        </Page>
        <Page>
          <Name>System</Name>
        </Page>
      </Pages>
    </StartRequestForm>
  </StartRequestForms>
</pam-ops:QueryStartRequestFormsResponse>
```

SubmitStartRequestForm

Para enviar um formulário de solicitação inicial, use os parâmetros definidos na resposta QueryStartRequestForms.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/SubmitStartRequestForm
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<SubmitStartRequestFormRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="p1:SubmitStartRequestFormRequest"

xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <Name>SRF</Name>
  <Path>/TestSRF</Path>
  <Arguments>
    <Argument>
      <Name>Var_0</Name>
      <Value>val0</Value>
    </Argument>
    <Argument>
      <Name>Var_1</Name>
      <Value>true</Value>
    </Argument>
    <Argument>
      <Name>Var_2</Name>
      <Value>val2</Value>
    </Argument>
  </Arguments>
  <Options>
    <StartDate>Data de início</StartDate>
    <StartTime>Hora de início</StartTime>
  </Options>
</SubmitStartRequestFormRequest>
```

Observação:

Use o seguinte formato para Data de início e Hora de início:

- Data de início: aaaa-MM-dd
- Hora de início: HH:mm (formato de 24 horas)

Conforme exibido no exemplo a seguir, a resposta de SubmitStartRequestForm contém o identificador de solicitação inicial (mostrado em **negrito** no exemplo):

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:SubmitStartRequestFormResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="pam-ops:SubmitStartRequestFormResponse">
  <StartRequestStatus>
    <InteractionID>1291</InteractionID>
    <Instance>SRF_1291</Instance>
    <State>Queued</State>

<Earliest-start-time>2012-06-14T15:52:05.341Z</Earliest-start-time
>
  <RefProcess>/TestSRF/TestProcess</RefProcess>
</StartRequestStatus>
</pam-ops:SubmitStartRequestFormResponse>
```

QueryStartRequests

Você pode consultar as solicitações iniciais usando um filtro opcional.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/QueryStartRequests
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<QueryStartRequestsRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:QueryStartRequestsRequest"

xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
<Instance>SRF_1291</Instance>
<IsArchived>>false</IsArchived>
</QueryStartRequestsRequest>
```

Como mostra o exemplo a seguir, a resposta de QueryStartRequests contém as instâncias da solicitação inicial:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:QueryStartRequestsResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="pam-ops:QueryStartRequestsResponse">
  <StartRequestInstances>
    <StartRequestInstance>
      <Instance>SRF_1291</Instance>
      <ProcessInstance>TestProcess_1292</ProcessInstance>
      <State>Completed</State>
      <Scheduled-time>2012-06-14T15:52:05.000Z</Scheduled-time>
      <Start-time>2012-06-14T15:52:05.000Z</Start-time>
      <End-time>2012-06-14T15:52:11.000Z</End-time>
    </StartRequestInstance>
  </StartRequestInstances>
</pam-ops:QueryStartRequestsResponse>
```

DequeueStartRequest

Você pode retirar uma solicitação inicial da fila.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/Dequeue
StartRequest
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<DequeueStartRequestRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="p1:DequeueStartRequestRequest"

xmlns:p1="http://ns.ca.com/2011/09/pam-ops">
  <Instance>SRF_1_44</Instance>
</DequeueStartRequestRequest>
```

O exemplo a seguir mostra a resposta de DequeueStartRequest:

```
<?xml version="1.0" encoding="UTF-8"?>
<pam-ops:DequeueStartRequestResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="pam-ops:DequeueStartRequestResponse">
  <StartRequestInstance>
    <Instance>SRF_1_44</Instance>
    <ProcessInstance xsi:nil="true"/>
    <State>Com falha</State>
    <Scheduled-time>2012-07-01T13:39:00.000Z</Scheduled-time>
  </StartRequestInstance>
</pam-ops:DequeueStartRequestResponse>
```

AbortStartRequest

Use AbortStartRequest para finalizar uma solicitação inicial em execução.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/AbortStartRequest
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<AbortStartRequestRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:AbortStartRequestRequest"

xmlns:p1="http://ns.ca.com/2011/09/pam-ops">
  <Instance>SRF_1_38</Instance>
</AbortStartRequestRequest>
```

O exemplo a seguir mostra a resposta de AbortStartRequest:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:AbortStartRequestResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="pam-ops:AbortStartRequestResponse">
  <StartRequestInstance>
    <Instance>SRF_1_38</Instance>
    <ProcessInstance>TestProcess_39</ProcessInstance>
    <State>Com falha</State>
    <Scheduled-time>2012-06-22T13:28:42.000Z</Scheduled-time>
    <Start-time>2012-06-22T13:28:42.000Z</Start-time>
  </StartRequestInstance>
</pam-ops:AbortStartRequestResponse>
```

Exemplos de operações REST de tarefas

Os exemplos a seguir demonstram como usar os serviços RESTful para realizar várias ações de tarefa.

QueryTasks

Você pode consultar uma tarefa específica ou todas as tarefas e solicitações de interação pendentes.

- Para consultar uma tarefa específica, forneça uma tag TaskID.
- Para consultar todas as tarefas, não especifique um qualificador TaskID.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/QueryTasks
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<QueryTasksRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="p1:QueryTasksRequest"
  xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <TaskID>510</TaskID>
</QueryTasksRequest>
```

A resposta contém a ID da tarefa, os nomes do parâmetro, os responsáveis e os usuários delegados. Use os nomes e os tipos dos parâmetros na operação ReplyTask (mostrado em **negrito** no exemplo).

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:QueryTasksResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="pam-ops:QueryTasksResponse">
```

Os tópicos a seguir descrevem como executar as tarefas mais rotineiras desta interface.

```
<Task>
  <TaskID>510</TaskID>
  <ProcessID>505</ProcessID>
  <State>Completed</State>
  <Instance>TestIRFPrompt_505</Instance>
  <Title>Assign title</Title>
  <Description>Assign description 1</Description>
  <StartTime>2012-06-01T17:37:08.000Z</StartTime>
  <RootUUID>eba34640-5de1-4e73-ab28-e312a1ef35fa</RootUUID>
  <IsApprovalRequired>true</IsApprovalRequired>
  <Pages>
    <HasComplexType>>false</HasComplexType>
    <Page>
      <Name>TestPage</Name>
      <Param>
        <Name>Var_0</Name>
        <Label>TextField1</Label>
        <Description/>
        <IsReadOnly>>false</IsReadOnly>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>Var_1</Name>
        <Label>Check Box</Label>
        <Description/>
        <IsReadOnly>>false</IsReadOnly>
        <Type>Boolean</Type>
      </Param>
      <Param>
        <Name>Var_2</Name>
        <Label>Select</Label>
        <Description/>
        <IsReadOnly>>false</IsReadOnly>
        <Type>String</Type>
      </Param>
    </Page>
    <Page>
      <Name>User Prompt</Name>
```

```
    </Page>
  <Page>
    <Name>System</Name>
  </Page>
</Pages>
<Assignees>
  <Users>
    <User>
      <Name>pamadmin</Name>
    </User>
  </Users>
</Assignees>
</Task>
</Tasks>
</pam-ops:QueryTasksResponse>
```

ReplyTask

Você pode responder a uma tarefa com valores de parâmetros e aprovação (mostrado em negrito no exemplo a seguir).

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/ReplyTask
```

Incluir um cabeçalho de solicitação:

Content-Type=application/xml

Por exemplo:

```
<ReplyTaskRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:ReplyTaskRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <TaskID>510</TaskID>
  <IsApproved>true</IsApproved>
  <Arguments>
    <Argument>
      <Name>Var_0</Name>
      <Value>val0</Value>
    </Argument>
    <Argument>
      <Name>Var_1</Name>
      <Value>>true</Value>
    </Argument>
    <Argument>
      <Name>Var_2</Name>
      <Value>val2</Value>
    </Argument>
  </Arguments>
</ReplyTaskRequest>
```

O exemplo a seguir mostra a resposta de ReplyTask:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:ReplyTaskResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="pam-ops:ReplyTaskResponse">
  <Task>
    <TaskID>510</TaskID>
    <ProcessID>505</ProcessID>
    <State>Completed</State>
    <Instance>TestIRFPrompt_505</Instance>
    <Title>Assign title</Title>
    <Description>Assign description 1</Description>
    <StartTime>2012-06-01T17:37:08.000Z</StartTime>
    <RootUUID>eba34640-5de1-4e73-ab28-e312a1ef35fa</RootUUID>
    <IsApprovalRequired>>true</IsApprovalRequired>
    <Pages>
      <HasComplexType>>false</HasComplexType>
      <Page>
        <Name>TestPage</Name>
        <Param>
          <Name>Var_0</Name>
          <Label>TextField1</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>String</Type>
        </Param>
        <Param>
          <Name>Var_1</Name>
          <Label>Check Box</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>Boolean</Type>
        </Param>
        <Param>
          <Name>Var_2</Name>
          <Label>Select</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>String</Type>
        </Param>
      </Page>
      <Page>
        <Name>User Prompt</Name>
      </Page>
      <Page>
        <Name>System</Name>
      </Page>
    </Pages>
  </Task>
</pam-ops:ReplyTaskResponse>
```

```
</Pages>
<Assignees>
  <Users>
    <User>
      <Name>pamadmin</Name>
    </User>
  </Users>
</Assignees>
<Delegates>
  <Users>
    <User>
      <Name>pamadmin</Name>
    </User>
  </Users>
</Delegates>
</Task>
</pam-ops:ReplyTaskResponse>
```

TakeTask

É possível executar uma tarefa por ID da tarefa.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/TakeTask
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<TakeTaskRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:TakeTaskRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <TaskID>510</TaskID>
</TakeTaskRequest>
```

A resposta:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:TakeTaskResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:type="pam-ops:TakeTaskResponse">
  <Task>
    <TaskID>510</TaskID>
    <ProcessID>505</ProcessID>
```

<State>eiamCertKeyPath

Especifica o nome do arquivo de chave do certificado de segurança usado para autenticação. Essa propriedade só será aplicável se isFipsMode=true.

Exemplo

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:TakeTaskResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:type="pam-ops:TakeTaskResponse">
  <Task>
    <TaskID>510</TaskID>
    <ProcessID>505</ProcessID>
    <State>Taken</State>
    <Instance>TestIRFPrompt_505</Instance>
    <Title>Assign title</Title>
    <Description>Assign description 1</Description>
    <StartTime>2012-06-01T17:37:08.000Z</StartTime>
    <RootUUID>eba34640-5de1-4e73-ab28-e312a1ef35fa</RootUUID>
    <IsApprovalRequired>true</IsApprovalRequired>
    <Pages>
      <HasComplexType>false</HasComplexType>
      <Page>
        <Name>TestPage</Name>
        <Param>
          <Name>Var_0</Name>
          <Label>TextField1</Label>
          <Description/>
          <IsReadOnly>false</IsReadOnly>
          <Type>String</Type>
        </Param>
        <Param>
          <Name>Var_1</Name>
          <Label>Check Box</Label>
          <Description/>
          <IsReadOnly>false</IsReadOnly>
          <Type>Boolean</Type>
```

```

        </Param>
        <Param>
          <Name>Var_2</Name>
          <Label>Select</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>String</Type>
        </Param>
      </Page>
      <Page>
        <Name>User Prompt</Name>
      </Page>
      <Page>
        <Name>System</Name>
      </Page>
    </Pages>
    <Assignees>
      <Users>
        <User>
          <Name>pamadmin</Name>
        </User>
      </Users>
    </Assignees>
  </Task>
</pam-ops:TakeTaskResponse>

```

ReturnTask

É possível retornar uma tarefa por ID da tarefa.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/ReturnTask
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```

<ReturnTaskRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:ReturnTaskRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <TaskID>510</TaskID>
</ReturnTaskRequest>

```

A resposta:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:ReturnTaskResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:type="pam-ops:ReturnTaskResponse">
  <Task>
    <TaskID>510</TaskID>
    <ProcessID>505</ProcessID>
    <State>Pending</State>
    <Instance>TestIRFPrompt_505</Instance>
    <Title>Assign title</Title>
    <Description>Assign description 1</Description>
    <StartTime>2012-06-01T17:37:08.000Z</StartTime>
    <RootUUID>eba34640-5de1-4e73-ab28-e312a1ef35fa</RootUUID>
    <IsApprovalRequired>true</IsApprovalRequired>
    <Pages>
      <HasComplexType>>false</HasComplexType>
      <Page>
        <Name>TestPage</Name>
        <Param>
          <Name>Var_0</Name>
          <Label>TextField1</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>String</Type>
        </Param>
        <Param>
          <Name>Var_1</Name>
          <Label>Check Box</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>Boolean</Type>
        </Param>
        <Param>
          <Name>Var_2</Name>
          <Label>Select</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>String</Type>
        </Param>
      </Page>
      <Page>
        <Name>User Prompt</Name>
      </Page>
      <Page>
        <Name>System</Name>
      </Page>
    </Pages>
  </Task>
</pam-ops:ReturnTaskResponse>
```

```
</Pages>  
<Assignees>
```

DelegateTask

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/DelegateTask
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<DelegateTaskRequest  
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
      xsi:type="p1:DelegateTaskRequest"  
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >  
  <TaskID>510</TaskID>  
  <Users>  
    <User>  
      <Name>pamadmin</Name>  
    </User>  
  </Users>  
  <Groups>  
    <Group/>  
  </Groups>  
</DelegateTaskRequest>
```

A resposta:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:DelegateTaskResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="pam-ops:DelegateTaskResponse">
  <Task>
    <TaskID>510</TaskID>
    <ProcessID>505</ProcessID>
    <State>Pending</State>
    <Instance>TestIRFPrompt_505</Instance>
    <Title>Assign title</Title>
    <Description>Assign description 1</Description>
    <StartTime>2012-06-01T17:37:08.000Z</StartTime>
    <RootUUID>eba34640-5de1-4e73-ab28-e312a1ef35fa</RootUUID>
    <IsApprovalRequired>true</IsApprovalRequired>
    <Pages>
      <HasComplexType>>false</HasComplexType>
      <Page>
        <Name>TestPage</Name>
        <Param>
          <Name>Var_0</Name>
          <Label>TextField1</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>String</Type>
        </Param>
        <Param>
          <Name>Var_1</Name>
          <Label>Check Box</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>Boolean</Type>
        </Param>
        <Param>
          <Name>Var_2</Name>
          <Label>Select</Label>
          <Description/>
          <IsReadOnly>>false</IsReadOnly>
          <Type>String</Type>
        </Param>
      </Page>
      <Page>
        <Name>User Prompt</Name>
      </Page>
    </Pages>
  </Task>
</pam-ops:DelegateTaskResponse>
```

TransferTask

É possível transferir (atribuir) uma tarefa por ID da tarefa para um ou mais usuários de um ou mais grupos.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/TransferTask
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<TransferTaskRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:TransferTaskRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <TaskID>510</TaskID>
  <Users>
    <User>
      <Name>pamadmin</Name>
    </User>
  </Users>
  <Groups>
    <Group/>
  </Groups>
</TransferTaskRequest>
```

A solicitação TransferTask retorna o seguinte XML:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:TransferTaskResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="pam-ops:TransferTaskResponse">
  <Task>
    <TaskID>510</TaskID>
    <ProcessID>505</ProcessID>
    <State>Pending</State>
    <Instance>TestIRFPrompt_505</Instance>
    <Title>Assign title</Title>
    <Description>Assign description 1</Description>
    <StartTime>2012-06-01T17:37:08.000Z</StartTime>
    <RootUUID>eba34640-5de1-4e73-ab28-e312a1ef35fa</RootUUID>
    <IsApprovalRequired>true</IsApprovalRequired>
    <Pages>
      <HasComplexType>false</HasComplexType>
      <Page>
        <Name>TestPage</Name>
        <Param>
          <Name>Var_0</Name>
          <Label>TextField1</Label>
          <Description/>
          <IsReadOnly>false</IsReadOnly>
          <Type>String</Type>
        </Param>
        <Param>
          <Name>Var_1</Name>
          <Label>Check Box</Label>
          <Description/>
          <IsReadOnly>false</IsReadOnly>
          <Type>Boolean</Type>
        </Param>
        <Param>
          <Name>Var_2</Name>
          <Label>Select</Label>
          <Description/>
          <IsReadOnly>false</IsReadOnly>
          <Type>String</Type>
        </Param>
      </Page>
      <Page>
        <Name>User Prompt</Name>
      </Page>
      <Page>
        <Name>System</Name>
      </Page>
    </Pages>
  </Task>
</pam-ops:TransferTaskResponse>
```

```
    </Page>
  </Pages>
  <Assignees>
    <Users>
      <User>
        <Name>pamadmin</Name>
      </User>
    </Users>
  </Assignees>
</Task>
</pam-ops:TransferTaskResponse>
```

Exemplos de conteúdo REST

As operações de conteúdo permitem exportar o conteúdo de um ambiente do CA Process Automation e importá-lo em outro ambiente do CA Process Automation.

Importar

Você pode importar um arquivo local de objetos do CA Process Automation.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/Import
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<ImportRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="pl:ImportRequest"
  xmlns:pl="http://ns.ca.com/2011/09/pam-ops" >
  <ImportLocation>TestContent</ImportLocation>
  <SourceLocation>C:\Arquivos de
programas\CA\PAM40\standalone\.c2orepository\public\content\ITPAMC
ontent.xml</SourceLocation>
  <OverwriteAction>ImportAndReplace</OverwriteAction>
  <SetCurrent>true</SetCurrent>
</ImportRequest>
```

Se a biblioteca já possuir um objeto com o nome especificado, defina <OverwriteAction> como uma das seguintes opções:

Importar

Aumenta a versão do objeto importado se existir um objeto de mesmo nome na pasta de importação especificada. Substitui a versão da release se ela já existir.

DoNotImport

Não importar objetos que tiverem o mesmo nome que um objeto existente.

Importar e substituir

Importar e substituir o objeto existente. Exclui as versões anteriores do objeto.

A resposta indica se a importação foi bem-sucedida:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:ImportResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="pam-ops:ImportResponse">
  <SuccessMessage>0 objeto especificado foi importado com
sucesso.</SuccessMessage>
</pam-ops:ImportResponse>
```

Exportar

Você pode exportar objetos do CA Process Automation para um arquivo local.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/Export
```

Incluir um cabeçalho de solicitação:

Content-Type=application/xml

Por exemplo:

```
<ExportRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="p1:ExportRequest"
xmlns:p1="http://ns.ca.com/2011/09/pam-ops">
  <FolderName>/rest</FolderName>
  <ExportAsContentPackage>true</ExportAsContentPackage>
  <IsAbsolute>true</IsAbsolute>
  <ExportFileName>
  <ExportLocation>C:\\REST\\ExportContent.xml</ExportLocation>
  <OverwriteFile>true</OverwriteFile>
</ExportFileName>
  <Filter>
  <ObjectTypes>
  <ObjectType><Type>Process</Type></ObjectType>
  <ObjectType><Type>Agenda</Type></ObjectType>
  <ObjectType><Type>ContentPackage</Type></ObjectType>
  </ObjectTypes>
  </Filter>
</ExportRequest>
```

Observação: o elemento SealModifiableReleaseVersions não é exigido e seu valor é implicitamente decidido com base no valor do elemento ExportAsContentPackage.

O sinalizador <ExportAsContentPackage> especifica se você pode exportar uma pasta como pacote de conteúdo. O valor padrão é falso.

Se você definir o sinalizador como verdadeiro, o CA Process Automation exportará a pasta como pacote de conteúdo.

Observação: se a versão da release da pasta e de seus objetos filho não estiver definida, a exportação de um pacote de conteúdo falhará.

A resposta indica se a exportação foi bem-sucedida:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:ExportResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="pam-ops:ExportResponse">
  <ExportMessage>Carregamento de biblioteca exportada feito com
sucesso para nome do arquivo para download:
C:\\REST\\ExportContent.xml</ExportMessage>
</pam-ops:ExportResponse>
```

Arquivos XML criptografados

Os métodos de serviço web de importação e exportação do Catalyst oferecem suporte a arquivos XML criptografados. Um arquivo XML exportado do CA Process Automation é criptografado se ele contiver objetos com um atributo de versão de release não modificável.

- É possível importar um arquivo em branco ou um arquivo XML criptografado.
- É possível exportar um arquivo em branco ou um arquivo XML criptografado. Se o CA Process Automation detectar que a versão de release de um dos arquivos exportados está bloqueada, ele criptografará todo o arquivo XML. Se o CA Process Automation detectar que as versões de release de todos os arquivos exportados estão desbloqueadas, ele exportará o arquivo XML sem criptografia.

Exemplos de REST de conjuntos de dados

É possível consultar e atualizar conjuntos de dados nomeados globais. Para executar as atualizações, use o código JavaScript da mesma maneira que usaria o código de execução anterior e posterior de um processo.

QueryDatasetNames

A função QueryDatasetNames permite consultar nomes de conjuntos de dados com um filtro opcional no caminho.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/QueryDatasetNames
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<QueryDatasetNamesRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:QueryDatasetNamesRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops"
>
    <PathName>/TestContent</PathName>
</QueryDatasetNamesRequest>
```

A resposta contém os nomes de caminho dos conjuntos de dados. O exemplo a seguir mostra os nomes de caminho em **negrito**:

```
<pam-ops:QueryDatasetNamesResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="pam-ops:QueryDatasetNamesResponse">
  <Datasets>
    <Dataset>
      <PathName>/TestContent/DatasetTest</PathName>
    </Dataset>
    <Dataset>
      <PathName>/TestContent/TestDataset</PathName>
    </Dataset>
    <Dataset>
      <PathName>/TestContent/TestDataset</PathName>
    </Dataset>
    <Dataset>
      <PathName>/TestContent/TestDataset_1</PathName>
    </Dataset>
  </Datasets>
</pam-ops:QueryDatasetNamesResponse>>
```

QueryDatasetParameters

Essa função pesquisa o caminho que você especificar entre as tags <PathName> e </PathName> na solicitação em relação a parâmetros de conjunto de dados.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/QueryDatasetParameters
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<QueryDatasetParametersRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:QueryDatasetParametersRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops"
>
  <PathName>/TestDatasets/TestDataset</PathName>
</QueryDatasetParametersRequest>
```

A resposta contém os parâmetros de conjunto de dados, incluindo seus nomes de JavaScript:

```
<pam-ops:QueryDatasetParametersResponse
  xmlns:pam-ops=http://ns.ca.com/2011/09/pam-ops
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="pam-ops:QueryDatasetParametersResponse">
  <Dataset>
    <PathName>/TestDatasets/TestDataset</PathName>
    <Parameters>
      <Param>
        <Param>
          <Name>Datasets["/TestDatasets/TestDataset"].Name</Name>
          <Value>John Doe</Value>
          <Label>Name</Label>
          <Type>String</Type>
        </Param>
      </Param>
      <Param>
        <Name>Datasets["/TestDatasets/TestDataset"].Address</Name>
        <Value>1400 Any Avenue</Value>
        <Label>Address</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>Datasets["/TestDatasets/TestDataset"].City</Name>
        <Value>DC</Value>
        <Label>City</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>Datasets["/TestDatasets/TestDataset"].Zipcode</Name>
        <Value>00000</Value>
        <Label>Zipcode</Label>
        <Type>Integer</Type>
      </Param>
      <Param>
        <Name>Datasets["/TestDatasets/TestDataset"].VMap.VarString</Name>
        <Value>updated1</Value>
        <Label>VarString</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>Datasets["/TestDatasets/TestDataset"].VMap.VarDate</Name>
        <Value xsi:nil="true"/>
        <Label>VarDate</Label>
        <Type>Date</Type>
      </Param>
    </Parameters>
  </Dataset>
</pam-ops:QueryDatasetParametersResponse>
```

```
</Dataset>  
</pam-ops:QueryDatasetParametersResponse>
```

QueryProcessDataset

Esta função pesquisa um conjunto de dados do processo. Especifique o nome da instância do processo que inclui o conjunto de dados que você deseja consultar na tag <InstanceName> na solicitação REST.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/QueryProcessDataset
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo, uma solicitação REST pode ser:

```
<QueryProcessDatasetRequest  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
    xsi:type="p1:QueryProcessDatasetRequest"  
    xmlns:p1="http://ns.ca.com/2011/09/pam-ops"  
>  
  <InstanceName>Process_388</InstanceName>  
</QueryProcessDatasetRequest>
```

A resposta retorna o conjunto de dados do processo na tag <ProcessDataset>. A resposta REST retorna os parâmetros que estiverem incluídos no conjunto de dados. Por exemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<pam-ops:QueryProcessDatasetResponse
xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="pam-ops:QueryProcessDatasetResponse">
  <ProcessDataset>
    <InstanceName>Process_388</InstanceName>
    <Parameters>
      <Param>
        <Name>ParentProcessROID</Name>
        <Value/>
        <Label>ParentProcessROID</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>RootProcessROID</Name>
        <Value/>
        <Label>ROID do processo raiz</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>Progress</Name>
        <Value>100.0</Value>
        <Label>Progress</Label>
        <Type>Double</Type>
      </Param>
      <Param>
        <Name>TouchpointName</Name>
        <Value>Orchestrator</Value>
        <Label>TouchpointName</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>ServerName</Name>
        <Value>SINRA22.ca.com</Value>
        <Label>ServerName</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>ServerId</Name>
        <Value>1ed56246-7880-4c10-8a5f-5a9975c17d9b</Value>
        <Label>ServerId</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>DisplayName</Name>
```

```
<Value>Process</Value>
<Label>DisplayName</Label>
<Type>String</Type>
</Param>
<Param>
<Name>InstanceUUID</Name>
<Value>024d68c3-1248-484b-aca1-ef8584492eac</Value>
<Label>InstanceUUID</Label>
<Type>String</Type>
</Param>
<Param>
<Name>rootUUID</Name>
<Value>024d68c3-1248-484b-aca1-ef8584492eac</Value>
<Label>rootUUID</Label>
<Type>String</Type>
</Param>
<Param>
<Name>CallerUser</Name>
<Value>pamadmin</Value>
<Label>CallerUser</Label>
<Type>String</Type>
</Param>
<Param>
<Name>effectiveUser</Name>
<Value>pamadmin</Value>
<Label>effectiveUser</Label>
<Type>String</Type>
</Param>
<Param>
<Name>RuntimeROID</Name>
<Value>388</Value>
<Label>RuntimeROID</Label>
<Type>Long</Type>
</Param>
<Param>
<Name>InstanceName</Name>
<Value>Process_388</Value>
<Label>InstanceName</Label>
<Type>String</Type>
</Param>
<Param>
<Name>ScheduledStartTime</Name>
<Value>14/01/13</Value>
<Label>ScheduledStartTime</Label>
<Type>Date</Type>
</Param>
<Param>
<Name>StartDate</Name>
<Value>14/01/13</Value>
```

```
<Label>StartDate</Label>
<Type>Date</Type>
</Param>
<Param>
<Name>StartTime</Name>
<Value>14/01/13</Value>
<Label>StartTime</Label>
<Type>Date</Type>
</Param>
<Param>
<Name>EndTime</Name>
<Value>14/01/13</Value>
<Label>EndTime</Label>
<Type>Date</Type>
</Param>
<Param>
<Name>EndDate</Name>
<Value>14/01/13</Value>
<Label>EndDate</Label>
<Type>Date</Type>
</Param>
<Param>
<Name>Result</Name>
<Value>1</Value>
<Label>Result</Label>
<Type>String</Type>
</Param>
</Parameters>
</ProcessDataset>
</pam-ops:QueryProcessDatasetResponse>
```

UpdateDatasetParameters

Para atualizar os parâmetros do conjunto de dados usando REST, especifique o nome do parâmetro na resposta QueryDatasetParameters. É possível usar expressões no valor.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/UpdatedDatasetParameters
```

Incluir um cabeçalho de solicitação:

Content-Type=application/xml

Por exemplo:

```
<UpdateDatasetParametersRequest

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:type="p1:UpdateDatasetParametersRequest"
xmlns:p1="http://ns.ca.com/2011/09/pam-ops"
>
  <Dataset>
    <PathName>/TestDatasets/TestDataset</PathName>
  <Parameters>
    <Param>
      <Name>Datasets["/TestDatasets/TestDataset"].Zipcode</Name>
      <Value>Datasets["/TestDatasets/TestDataset"].Zipcode +
100</Value>
      <Type>Integer</Type>
    </Param>
  </Parameters>
</Dataset>
</UpdateDatasetParametersRequest>
```

A resposta contém o conjunto de dados atualizado:

```
<?xml version="1.0" encoding="UTF-8"?>
<pam-ops:UpdateDatasetParametersResponse
  xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="pam-ops:UpdateDatasetParametersResponse">
  <Dataset>
    <PathName>/TestDatasets/TestDataset</PathName>
    <Parameters>
      <Param>
        <Name>Datasets["/TestDatasets/TestDataset"].Name</Name>
        <Value>Harry Truman</Value>
        <Label>Name</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>Datasets["/TestDatasets/TestDataset"].Address</Name>
        <Value>1400 Pennsylvania Avenue</Value>
        <Label>Address</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>Datasets["/TestDatasets/TestDataset"].City</Name>
        <Value>DC</Value>
        <Label>City</Label>
        <Type>String</Type>
      </Param>
      <Param>
        <Name>Datasets["/TestDatasets/TestDataset"].Zipcode</Name>
        <Value>2300</Value>
        <Label>Zipcode</Label>
        <Type>Integer</Type>
      </Param>
    </Parameters>
  </Dataset>
</pam-ops:UpdateDatasetParametersResponse>
```

Exemplos de REST de configuração de módulo

As operações de configuração de módulo permitem consultar e atualizar módulos do CA Process Automation da mesma maneira que na guia Módulos do Navegador de configuração.

QueryModuleConfigs

Para consultar nomes de módulo, use a função QueryModuleConfigs.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/QueryModuleConfigs
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

O exemplo a seguir retornará todos os módulos:

```
<QueryModuleConfigsRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:QueryModuleConfigsRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <ModuleName xsi:nil="true"/>
  (Opcional) <ModuleType>ALL</ModuleType>
</QueryModuleConfigsRequest>
```

O exemplo a seguir retorna os módulos personalizados:

```
<QueryModuleConfigsRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:type="p1:QueryModuleConfigsRequest"
      xmlns:p1="http://ns.ca.com/2011/09/pam-ops" >
  <ModuleName xsi:nil="true"/>
  (Opcional) <ModuleType>CUSTOM</ModuleType>
</QueryModuleConfigsRequest>
```

A resposta contém os nomes do módulo:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:QueryModuleConfigsResponse
  xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"

  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="pam-ops:QueryModuleConfigsResponse">
  <Module>
    <ModuleName>FTPServices</ModuleName>
    <Name>File Transfer</Name>
    <DisplayName>File Transfer</DisplayName>
    <Description>Provides file transfer operations
(FTP/SFTP).</Description>
  </Module>
  <Module>
    <ModuleName>WorkflowServices</ModuleName>
    <Name>Process Control</Name>
    <DisplayName>Process Control</DisplayName>
    <Description>Runs, monitors and controls CA Process Automation
Processes.</Description>
  </Module>
  <Module>
    <ModuleName>UtilitiesGroup</ModuleName>
    <Name>Utilities</Name>
    <DisplayName>Utilities</DisplayName>
    <Description>This module consists of utility operators which are
used in PAM processes</Description>
  </Module>
  <Module>
    <ModuleName>SOAP Services</ModuleName>
    <Name>Web Services</Name>
    <DisplayName>Web Services</DisplayName>
    <Description>Provides an interface to external services exposed
through SOAP.</Description>
  </Module>
  <Module>
    <ModuleName>JMX.Group.Name</ModuleName>
    <Name>Java Management</Name>
    <DisplayName>Java Management</DisplayName>
    <Description>Provides a management interface to external system
that support JMX.</Description>
  </Module>
  <Module>
    <ModuleName>Process Services</ModuleName>
    <Name>Command Execution</Name>
    <DisplayName>Command Execution</DisplayName>
    <Description>Runs programs and scripts on host operating
environments.</Description>
  </Module>
```

```
<Module>
  <ModuleName>Date Time Services</ModuleName>
  <Name>Date-Time</Name>
  <DisplayName>Date-Time</DisplayName>
  <Description>Executes time and calendar constraints in CA Process
Automation processes.</Description>
</Module>
<Module>
  <ModuleName>FileService</ModuleName>
  <Name>File Management</Name>
  <DisplayName>File Management</DisplayName>
  <Description>This module monitors directory, files, and their
contents</Description>
</Module>
```

QueryModuleConfigProperties

Para consultar as propriedades do módulo, use a função QueryModuleConfigProperties.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/QueryModuleConfigProperties
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<QueryModuleConfigPropertiesRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="p1:QueryModuleConfigPropertiesRequest"
          xmlns:p1="http://ns.ca.com/2011/09/pam-ops"
>
  <ModuleName>WorkflowServices</ModuleName>
</QueryModuleConfigPropertiesRequest>
```

A resposta contém as propriedades do módulo:

```
<?xml version="1.0" encoding="UTF-8"?>
<pam-ops:QueryModuleConfigPropertiesResponse
  xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="pam-ops:QueryModuleConfigPropertiesResponse">
  <ModuleName>WorkflowServices</ModuleName>
  <Property>
    <Name>TimeToKeepPrompts</Name>
    <Value>5</Value>
    <Label>Time to keep completed user interactions (mins)</Label>
    <Description>This parameter determines the maximum time in
minutes that a particular prompt information remains accessible to the
user once the prompt has ended.</Description>
    <Page>Default Process Control Properties</Page>
    <Type>Long</Type>
  </Property>
</pam-ops:QueryModuleConfigPropertiesResponse>
```

UpdateModuleConfigProperties

Para atualizar as propriedades de um módulo, use a função UpdateModuleConfigProperties.

Por exemplo, envie uma solicitação HTTP POST para:

```
http://<hostname>:7000/node/rest/CA:00074_CA:00074:01/_ops/UpdateModuleConfigProperties
```

Incluir um cabeçalho de solicitação:

```
Content-Type=application/xml
```

Por exemplo:

```
<UpdateModuleConfigPropertiesRequest
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:type="p1:UpdateModuleConfigPropertiesRequest"
  xmlns:p1="http://ns.ca.com/2011/09/pam-ops"
>
  <ModuleName>WorkflowServices</ModuleName>
  <Property>
    <Name>TimeToKeepPrompts</Name>
    <Value>4</Value>
    <Type>Long</Type>
  </Property>
</UpdateModuleConfigPropertiesRequest>
```

A resposta contém as propriedades do módulo atualizadas:

```
<?xml version="1.0" encoding="utf-8"?>
<pam-ops:UpdateModuleConfigPropertiesResponse
  xmlns:pam-ops="http://ns.ca.com/2011/09/pam-ops"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:type="pam-ops:UpdateModuleConfigPropertiesResponse">
  <Property>
    <Name>TimeToKeepPrompts</Name>
    <Value>4</Value>
    <Type>Long</Type>
  </Property>
</pam-ops:UpdateModuleConfigPropertiesResponse>
```

Utilização de OData do Catalyst

OData (Open Data Protocol) é um protocolo da web para consulta e atualização de dados. O OData fornece uma maneira para desbloquear seus dados e liberá-los de silos de aplicativos. O OData faz isso aplicando e criando sobre tecnologias da web, como HTTP, AtomPub (Atom Publishing Protocol) e JSON, a fim de fornecer acesso às informações de vários aplicativos, serviços e armazenamentos. O OData expõe e acessa informações de diversas fontes, incluindo bancos de dados relacionais, sistemas de arquivos, sistemas de gerenciamento de conteúdo e websites tradicionais.

Observação: para obter mais informações sobre OData, consulte <http://www.odata.org/>.

O OData adiciona as seguintes funcionalidade ao AtomPub:

- Uma convenção para representar dados estruturados.
- Um recurso que aborda esquemas e sintaxes de URL.
- Opções de consulta comuns, como filtragem e classificação.
- Um esquema que descreve a estrutura, os links e os metadados de recursos.
- Formatos e semânticas de carga para solicitações em lote e unidade de trabalho.
- Representações alternativas de conteúdo de recurso (JSON).

O URL do OData do Catalyst tem o seguinte formato geral:

```
http://<hostname>:7000/node/odata/<module>
```

O URL do OData segue a mesma sintaxe que os URLs de REST, exceto pelo fato de você especificar "odata" em vez de "rest". O formato padrão é AtomPub. Para recuperar um formato JSON, anexe "?\$format=json" ao URL.

Você também pode especificar filtros de consulta de OData. Por exemplo, um código semelhante ao seguinte consulta processos cancelados:

```
http://<hostname>:7000/node/odata/CA:00074_CA:00074:01/ITActivity?$  
filter=StateDescription eq 'Aborted'
```

Observação: para obter mais informações sobre a especificação de OData, consulte <http://www.odata.org>.

Capítulo 3: Referência da API SOAP

O SOAP (Simple Object Access Protocol) é uma especificação de protocolo para troca de informações estruturadas na implementação de serviços web.

O WSDL para a interface de serviços web do CA Process Automation pode ser acessado em:

`http(s)://<CA Process Automation Server Name>:<port>/itpam/soap?wsdl`

Observações

- A entrada de serviços web do CA Process Automation oferece suporte apenas para o SOAP versão 1.1.
- Os métodos de serviços web não serão estendidos além do CA Process Automation Release 04.1.00. A documentação é mantida para fins de referência somente.

Esta seção contém os seguintes tópicos:

[Métodos de serviços web](#) (na página 109)

[Tags comuns para métodos de serviços web](#) (na página 172)

Métodos de serviços web

Para executar uma chamada de SOAP, você pode usar um token ou uma combinação de nome de usuário e senha.

O exemplo a seguir mostra um nome de usuário/senha de autenticação e autorização:

```
CA AuthMinder
  <user>Joe</ user >
  <password>thisismypassword</ password >
</auth>
```

Observação: consulte Tags comuns para métodos de serviços web para aprender sobre tags comuns que se aplicam a vários métodos de serviços web.

AsyncSoapResponse

Use este método para concluir um operador Chamar método SOAP assíncrono que está em execução em qualquer instância de processo.

Exemplo

```
<tns:AsyncSoapResponse xmlns:tns="http://www.ca.com/itpam">
<tns:MessageID>514514e3-e8c8-4b1d-be42-3ee85e9d8d37</tns:MessageID>
<!--Any valid XML fragment-->
</tns:AsyncSoapResponse>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:AsyncSoapResponse xmlns:tns="http://www.ca.com/itpam">
<tns:MessageID>514514e3-e8c8-4b1d-be42-3ee85e9d8d37</tns:MessageID>
<!--Any valid XML fragment-->
</tns:AsyncSoapResponse></SOAP-ENV:Body></SOAP-ENV:Envelope>
```

<MessageID> especifica a UUID do operador Chamar método SOAP assíncrono que está em execução.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<AsyncSoapResponseResponse xmlns="http://www.ca.com/itpam">
<status>SOAP_ASYNC_SUCCESS</status>
</AsyncSoapResponseResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

checkServerStatus

Use esse método para ver o status do servidor.

Exemplo

```
<tns:checkServerStatus xmlns:tns="http://www.ca.com/itpam">
  <tns:auth>
    <!--xsd:Choice Type-->
    <tns:token>token__</tns:token>
    <tns:user>user__</tns:user>
    <tns:password>password__</tns:password>
  </tns:auth>
</tns:checkServerStatus>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body><tns:checkServerStatus xmlns:tns="http://www.ca.com/itpam">
  <tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>pamadmin</tns:user>
<tns:password>pamadmin</tns:password>
  </tns:auth>
</tns:checkServerStatus>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<checkServerStatusResponse xmlns="http://www.ca.com/itpam">
<serverStatus>Status do servidor ok.</serverStatus>
</checkServerStatusResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

<serverStatus> mostra o status do servidor.

checkStartRequestStatus

Use este método para visualizar o status de uma solicitação inicial.

Exemplo

```
<tns:checkStartRequestStatus xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<!--xsd:Choice Type-->
<tns:interactionId>51</tns:interactionId>
<tns:tagId>tagId__</tns:tagId>
</tns:checkStartRequestStatus>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:checkStartRequestStatus xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<!--xsd:Choice Type-->
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:interactionId>51</tns:interactionId>
<tns:tagId>z34dsf5c7-xagb-4g5d-74g5-bd5c4we2378f</tns:tagId>
</tns:checkStartRequestStatus>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Você pode incluir um dos itens a seguir:

<*interactionId*>: o usuário passa a ID do formulário de solicitação inicial para exibição.

<*tagId*>: o usuário passa a tagId do formulário de solicitação inicial para exibição.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<checkStartRequestStatusResponse xmlns="http://www.ca.com/itpam">
<startRequestStatus>
<interactionId>51</interactionId>
<state>Completed</state>
<earliest-start-time>2012-03-01T12:51:25.690+05:30</earliest-start-time>
<start-time>2012-03-01T12:51:25.893+05:30</start-time>
<end-time>2012-03-01T12:51:38.033+05:30</end-time>
<refProcess>/TENSU03/Process</refProcess>
<processID>52</processID>
</startRequestStatus>
</checkStartRequestStatusResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

A resposta contém tags nas quais o usuário pode ver o resultado.

`<state>` descreve o estado da solicitação inicial.

`<earliest-start-time>` declara a hora em que a solicitação inicial foi colocada na fila.

`<start-time>` declara a hora de início real da solicitação inicial.

`<end-time>` declara a hora de finalização da solicitação inicial.

`<refProcess>` mostra o caminho de referência do processo anexado à solicitação inicial.

`<processID>` mostra a ID da instância do processo (somente retornado se uma instância do processo tiver sido criada).

controlInstance

Use este método para criar uma solicitação SOAP para controlar uma instância. O método `controlInstance` especifica o arquivamento para o processo ou para formulários de solicitação inicial (e todas as suas instâncias filho) que não foram acionados no modo desvinculado. Esse método não envia uma falha se um processo ou formulário de solicitação inicial já estiver marcado para arquivamento. O método `controlInstance` envia uma falha quando um processo ou formulário de solicitação inicial não tiver sido encontrado com a ID especificada.

Para executar um método `controlInstance`, use qualquer uma das seguintes opções:

- `rootUUID`: essa opção especifica a `rootUUID` do processo ou formulário de solicitação inicial
- `instanceId`: essa opção especifica a ID da instância do processo ou formulário de solicitação inicial
- `tagId`: essa opção especifica a `tagId` do processo ou formulário de solicitação inicial.

Exemplo

```
<tns:controlInstance xmlns:tns="http://www.ca.com/itpam">
  <tns:auth>
    <!--xsd:Choice Type-->
    <tns:token>token__</tns:token>
    <tns:user>user__</tns:user>
    <tns:password>password__</tns:password>
  </tns:auth>
  <!--xsd:Choice Type-->
  <tns:instanceId>instanceId__</tns:instanceId>
  <tns:action>arquivar</tns:action>
</tns:controlInstance>
```

Solicitação de SOAP

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:itp="http://www.ca.com/itpam">
  <soapenv:Header/>
  <soapenv:Body>
    <tns:controlInstance>
      <tns:auth>
        <!--xsd:Choice Type-->
        <tns:token>token__</tns:token>
        <tns:user>Joe</tns:user>
        <tns:password>thisismypassword</tns:password>
      <tns:auth>
        <!--xsd:Choice Type-->
        <tns:instanceId>3186</tns:instanceId>
        <tns:action>arquivar</tns:action>
      </tns:controlInstance>
    </soapenv:Body>
  </soapenv:Envelope>
```

<rootUUID> especifica o rootUUID do processo/formulário de solicitação inicial.

<instanceId> especifica a ID da instância do processo/formulário de solicitação inicial.

<tagId> especifica o tagId do processo/formulário de solicitação inicial.

<action> especifica a ação a ser executada no processo/formulário de solicitação inicial. No momento, só há suporte para arquivar.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <controlProcessResponse xmlns="http://www.ca.com/itpam">
      <actionStatus>A ação arquivar foi executada com êxito.</actionStatus>
    </controlProcessResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

controlProcess

Use esse método para criar uma solicitação para controlar um processo.

Exemplo

```
<tns:controlProcess xmlns:tns="http://www.ca.com/itpam">
  <tns:ProcessID>ProcessID__</tns:ProcessID>
  <tns:action>action__</tns:action>
  <tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>user__</tns:user>
<tns:password>password__</tns:password>
  </tns:auth>
</tns:controlProcess>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?><SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<SOAP-ENV:Header/>

<SOAP-ENV:Body>

<tns:controlProcess xmlns:tns="http://www.ca.com/itpam">
<tns:ProcessID>47</tns:ProcessID>
<tns:action>suspender</tns:action>
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
</tns:controlProcess></SOAP-ENV:Body></SOAP-ENV:Envelope>
```

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<controlProcessResponse xmlns="http://www.ca.com/itpam">
<actionStatus>A ação suspender para a ID de processo "47" estava em
fila.</actionStatus>
</controlProcessResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Ações

Suspende

Você pode suspender uma única instância de processo por meio de um método de serviços web.

Se não for possível suspender uma instância de processo (por exemplo, ela já foi concluída ou cancelada), será retornada uma mensagem de erro contendo os detalhes sobre a solicitação.

Este é um exemplo de uma falha ao tentar suspender um processo concluído:

```
<Fault xmlns="http://schemas.xmlsoap.org/soap/envelope/">  
<faultcode>SOAP-ENV:Server</faultcode>  
<faultstring>The process with Process ID "208" is in completed state and  
therefore cannot be suspended.</faultstring>  
</Fault>
```

Retomar

Você pode retomar uma única instância de processo que tenha sido anteriormente suspensa por meio de um método de serviços web.

Se não for possível retomar uma instância de processo, será retornada uma mensagem de erro contendo os detalhes sobre a solicitação. Por exemplo, uma instância de processo não pode ser retomada porque ela já foi concluída, cancelada ou não está suspensa no momento.

Este é um exemplo de uma falha ao tentar retomar um processo concluído:

```
<Fault xmlns="http://schemas.xmlsoap.org/soap/envelope/">  
<faultcode>SOAP-ENV:Server</faultcode>  
<faultstring>The process with Process ID "208" is in completed state and  
therefore cannot be resumed.</faultstring>  
</Fault>
```

Anular

Você pode anular uma única instância de processo por meio de um método de serviços web.

Se não for possível anular uma instância de processo (por exemplo, ela já foi concluída ou cancelada), será retornada uma mensagem de erro contendo os detalhes sobre a solicitação.

Este é um exemplo de uma falha ao tentar cancelar um processo concluído:

```
<Fault xmlns="http://schemas.xmlsoap.org/soap/envelope/">  
<faultcode>SOAP-ENV:Server</faultcode>  
<faultstring>The process with Process ID "486" is in completed state and  
therefore cannot be Aborted.</faultstring>  
</Fault>
```

Exemplo 1

```
<tns:controlProcess xmlns:tns="http://www.ca.com/itpam">
  <tns:ProcessID>ProcessID </tns:ProcessID>
  <tns:action>action</tns:action>
  <tns:auth>
    <!--xsd:Choice Type-->
    <tns:token>token__</tns:token>
    <tns:user>Joe</tns:user>
    <tns:password>thisismypassword</tns:password>
  </tns:auth></tns:controlProcess>
```

"*ProcessedID*" é a ROID da instância de processo que você deve controlar (suspender/retomar/cancelar). A "ROID" de um processo em execução está na resposta ao executar um processo por meio de serviços web.

Exemplo 2

```
<tns:ExecuteC20FlowResponse xmlns:tns="http://www.ca.com/itpam">
  <tns:ExecuteC20FlowResult>
    <tns:ROID>567</tns:ROID>
    ...
    ...
  </tns:ExecuteC20FlowResult>
</tns:ExecuteC20FlowResponse>
```

O valor para a ação é suspender, retomar ou cancelar. Somente um usuário autorizado pode executar esta ação; caso contrário, uma falha é retornada.

Exemplo 3

```
<SOAP-ENV:Fault xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring>Invalid user/password or token.</faultstring>
</SOAP-ENV:Fault>
```

Uma falha é retornada se você tentar executar uma operação que não pode ser executada em um processo em seu estado atual. Por exemplo, tentar suspender um processo que já foi concluído.

Exemplo 4

```
<SOAP-ENV:Fault xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring>Processo com a Id de processo "576" está em estado finalizado,
  portanto , não pode ser suspensa.</faultstring>
</SOAP-ENV:Fault>
```

deleteArchivedInstances

Use esse método para excluir as instâncias arquivadas do orquestrador do CA Process Automation.

Exemplo

```
<tns:deleteArchivedInstances xmlns:tns="http://www.ca.com/itpam">
  <tns:auth>
    <!--xsd:Choice Type-->
    <tns:token>token__</tns:token>
    <tns:user>Joe</tns:user>
    <tns:password>thisismypassword</tns:password>
  </tns:auth>
  <tns:dateRange>
    <tns:fromDate>2012-02-02T02:00:00,320+05:30</tns:fromDate>
    <tns:toDate>2012-02-02T03:00:00,320+05:30</tns:toDate>
  </tns:dateRange>
</tns:deleteArchivedInstances>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header/><SOAP-ENV:Body>
    <tns:deleteArchivedInstances xmlns:tns="http://www.ca.com/itpam">
      <tns:auth>
        <tns:user>Joe</tns:user>
        <tns:password>thisismypassword</tns:password>
      </tns:auth>
      <tns:dateRange>
        <tns:fromDate>2012-02-02T02:00:00,320+05:30</tns:fromDate>
        <tns:toDate>2012-02-02T03:00:00,320+05:30</tns:toDate>
      </tns:dateRange>
    </tns:deleteArchivedInstances>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Você pode especificar o *<dateRange>* nas tags *<fromDate>* e *<toDate>* para excluir as instâncias arquivadas. Use o formato padrão de XSD 2002-05-30T09:00:00.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<deleteArchivedInstancesResponse xmlns="http://www.ca.com/itpam">
<successMessage>4 archived instances were deleted successfully.</successMessage>
</deleteArchivedInstancesResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

<successMessage> mostra a exclusão bem-sucedida de instâncias e o número de instâncias excluídas.

Quando não houver instâncias no período especificado, a resposta será:

```
<deleteArchivedInstancesResponse xmlns="http://www.ca.com/itpam">
<successMessage>No archived instance is available for the specified date
range.</successMessage>
</deleteArchivedInstancesResponse>
```

deleteAttachments

Use esse método para excluir anexos que foram transferidos por upload, usando o método executeStartRequest/ExecuteProcess.

Exemplo

```
<tns:deleteAttachments xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:attachments>
<tns:attachmentId>111</tns:attachmentId>
</tns:attachments>
</tns:deleteAttachments>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:deleteAttachments xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:attachments>
<tns:attachmentId>111</tns:attachmentId>
</tns:attachments>
</tns:deleteAttachments>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<deleteAttachmentsResponse xmlns="http://www.ca.com/itpam">
<successMessage>The specified attachments were deleted
successfully.</successMessage>
</deleteAttachmentsResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

A tag *<successMessage>* mostra os resultados da solicitação.

executePendingInteraction

Use este método para executar as interações do usuário pendentes.

Exemplo

```
<tns:executePendingInteraction xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:params>
<tns:param name="Var_0">olá</tns:param>
<tns:param name="Var_1">mundo</tns:param>
</tns:params>
<tns:userInteractionID>22141</tns:userInteractionID>
<tns:isApprove>>true</tns:isApprove>
</tns:executePendingInteraction>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:executePendingInteraction xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:params>
<tns:param name="Var_0">olá</tns:param>
<tns:param name="Var_1">mundo</tns:param>
</tns:params>
<tns:userInteractionID>22141</tns:userInteractionID>
<tns:isApprove>>true</tns:isApprove>
</tns:executePendingInteraction>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Para fornecer um parâmetro

Na solicitação anterior, o usuário pode fornecer uma lista de parâmetros na tag <params>. O usuário deve criar uma tag <param> que inclui os seguintes parâmetros:

Nomeie o atributo dessa tag

Forneça o nome do parâmetro.

O valor dessa tag

Forneça o valor do parâmetro.

`<userInteractionID>` é a ID da tarefa da interação do usuário a ser executada.

`<isApprove>` especifica a aprovação ou rejeição da tarefa. Esse valor pode ser verdadeiro ou falso. Esta tag é opcional.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<executePendingInteractionResponse xmlns="http://www.ca.com/itpam">
<userInteractionID>22141</userInteractionID>
<interactionTitle>Task_MyTasks</interactionTitle>
<state>Completed</state>
<startTime>2012-02-29T18:11:28.817+05:30</startTime>
<endTime>2012-02-29T18:17:27.707+05:30</endTime>
<parmittedUserGroup>pamadmin</parmittedUserGroup>
<closedBy>pamadmin</closedBy>
</executePendingInteractionResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

`<userInteractionID>` especifica a ID da solicitação de interação.

`<interactionTitle>` especifica o título da solicitação de interação.

`<state>` especifica o estado da solicitação de interação.

`<startTime>` especifica a hora de início da solicitação de interação.

`<endTime>` especifica a hora de finalização da solicitação de interação.

`<parmittedUserGroup>` mostra a lista de usuários e grupos atribuídos.

`<closedBy>` mostra o nome de usuário que respondeu à tarefa.

Observação: o sinalizador isApprove estava sendo ignorado por esse método no CA Process Automation r2.1. Para corrigir isso, além de permitir a retenção de comportamento existente das instâncias existentes do CA Process Automation, uma nova instalação tem oasis.reject.unnecessary.approval definido como verdadeiro no arquivo de configuração OasisConfig.properties. As instalações existentes têm oasis.reject.unnecessary.approval adicionado para definir o arquivo de configuração OasisConfig.properties e definir como falso.

Se oasis.reject.unnecessary.approval for verdadeiro, o servidor considera o sinalizador isApprove. Uma falha é retornada se o usuário tentar aprovar ou rejeitar um formulário que não exigir aprovação. Uma falha também é retornada se a mensagem SOAP não incluir o sinalizador isApprove para um formulário que exigir aprovação.

executeProcess

Use este método para executar um processo.

Exemplo

```
<tns:executeProcess xmlns:tns="http://www.ca.com/itpam">
  <tns:flow>
    <tns:name>/MIMETEST</tns:name>
    <tns:action>iniciar</tns:action>
    <tns:auth>
      <!--xsd:Choice Type-->
      <tns:token>token__</tns:token>
      <tns:user>Joe</tns:user>
      <tns:password>thisismypassword</tns:password>
    </tns:auth>
    <tns:params>
      <tns:param name="name_6__">param__</tns:param>
    </tns:params>
    <tns:options>
      <tns:startDate>startDate__</tns:startDate>
      <tns:startTime>startTime__</tns:startTime>
      <tns:tagId> 06a4d113-0333-4aba-8cce-8781c18647c9</tns:tagId>
      <tns:autoArchive>true</tns:autoArchive>
      <tns:priority>priority__</tns:priority>
    </tns:options>
    <tns:attachments attachmentsParamName="nome">
      <tns:attachment>
        <tns:attachmentID>123</tns:attachmentID>
        <tns:name>MIMETEST</tns:name>
        <tns:localSourceLocation>"C:\\setupdir.log"</tns:localSourceLocation>
      </tns:attachment>
    </tns:attachments>
  </tns:flow>
</tns:executeProcess>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:executeProcess xmlns:tns="http://www.ca.com/itpam">
<tns:flow>
<tns:name>/MIMETEST</tns:name>
<tns:action>iniciar</tns:action>
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:params>
<tns:param name="name_6__">param__</tns:param>
</tns:params>
<tns:options>
<tns:startDate>startDate__</tns:startDate>
<tns:startTime>startTime__</tns:startTime>
<tns:tagId> 06a4d113-0333-4aba-8cce-8781c18647c9</tns:tagId>
<tns:isAutoArchive>true</tns:isAutoArchive>
<tns:priority>priority__</tns:priority>
</tns:options>
<tns:attachments attachmentsParamName="TENSU03">
<tns:attachment>
<tns:attachmentID>123</tns:attachmentID>
<tns:name>MIMETEST</tns:name>
<tns:localSourceLocation>"C:\\setupdir.log"</tns:localSourceLocation>
</tns:attachment>
</tns:attachments>
</tns:flow>
</tns:executeProcess>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

As tags a seguir estão sob a tag <flow>:

- <name> especifica o nome do processo, incluindo o caminho absoluto que o usuário deseja executar.
- <action> Especifica o "início" para iniciar o processo.
- <tagId> Adiciona um ID exclusivo a um processo/formulário de solicitação inicial. Uma falha retorna se houver uma instância de processo ou de formulário de solicitação inicial com a mesma tag. A tag <tagId> monitora o status de qualquer processo/formulário de solicitação inicial para o qual você especificar a tag.
- A tag <IsAutoArchive> pode ser definida como falso para ignorar instâncias de processo/formulário de solicitação inicial (juntamente com suas instâncias filho) do arquivamento na política de arquivamento. Processos que são executados em modo desvinculado não são afetados por essa tag. O valor padrão para <IsAutoArchive> é verdadeiro.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<ExecuteC20FlowResponse xmlns="http://www.ca.com/itpam">
  <ExecuteC20FlowResult>
    <ROID>150</ROID>
    <flow-name>/MIMETEST</flow-name>
    <flow-action>iniciar</flow-action>
    <auth-user>pamadmin</auth-user>
    <submission-time>2012-01-18T09:26:36.453+05:30</submission-time>
    <sender-address>nenhum</sender-address>
    <connection-id>nenhum</connection-id>
    <message>Documentos aceitos para processamento</message>
    <params>
      <param name="RuntimeROID">150</param>
      <param name="ServerId">1b4bda3d-bc5c-4a89-a2ad-23e2d4e71bce</param>
      <param name="effectiveUser">pamadmin</param>
      <param name="InstanceUUID">659322db-deca-4827-b622-14f69d09b33c</param>
      <param name="name_6__">param__</param>
      <param name="UserName">pamadmin</param>
      <param name="ProcessName">/MIMETEST</param>
      <param name="ServerName">lodivsa204.ca.com</param>
      <param name="DisplayName">MIMETEST</param>
      <param name="ProcessPriority">priority__</param>
      <param name="TouchpointName">Orquestrador</param>
      <param name="CallerUser">pamadmin</param>
      <param name="ScheduledStartTime">2012-01-18 00:00:00</param>
      <param name="rootUUID">659322db-deca-4827-b622-14f69d09b33c</param>
      <param name="ProcessAction">start</param>
      <param name="InstanceName">MIMETEST_150</param>
    </params>
  </ExecuteC20FlowResult>
</ExecuteC20FlowResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Nessa resposta, o usuário pode ver os detalhes da instância do processo que é executado.

<ROID> especifica o ROID da instância do processo.

executeStartRequest

Use este método para iniciar um formulário de solicitação inicial da biblioteca do CA Process Automation.

Exemplo

```
<tns:executeStartRequest xmlns:tns="http://www.ca.com/itpam">
  <tns:auth>
    <!--xsd:Choice Type-->
    <tns:token>token__</tns:token>
    <tns:user>Joe</tns:user>
    <tns:password>thisismypassword</tns:password>
  </tns:auth>
  <tns:objLocation>
    <tns:name>MYSRF</tns:name>
    <tns:path>/myfolder/</tns:path>
  </tns:objLocation>
  <tns:params>
    <tns:param name="Var_0">joe</tns:param>
    <tns:param name="Var_1">smith</tns:param>
  </tns:params>
  <tns:options>
    <tns:startDate>startDate__</tns:startDate>
    <tns:startTime>startTime__</tns:startTime>
    <tns:tagId> 06a4d113-0333-4aba-8cce-8781c18647c9</tns:tagId>
    <tns:isAutoArchive>true</tns:isAutoArchive>
    <tns:priority>priority__</tns:priority>
  </tns:options>
  <tns:attachments attachmentsParamName="attachmentsParamName__">
    <tns:attachment>
      <tns:attachmentID>attachmentID__</tns:attachmentID>
      <tns:name>name_11__</tns:name>
      <tns:localSourceLocation>localSourceLocation__</tns:localSourceLocation>
    </tns:attachment>
  </tns:attachments>
</tns:executeStartRequest>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:executeStartRequest xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:objLocation>
<tns:name>MYSRF</tns:name>
<tns:path>/TENSU03</tns:path>
</tns:objLocation>
<tns:params>
<tns:param name="Var_0">usuário</tns:param>
<tns:param name="Var_1">joe</tns:param>
</tns:params>
<tns:options>
<tns:startDate>startDate__</tns:startDate>
<tns:startTime>startTime__</tns:startTime>
<tns:tagId>06a4d113-0333-4aba-8cce-8781c18647c9</tns:tagId>
<tns:isAutoArchive>true</tns:isAutoArchive>
<tns:priority>priority__</tns:priority>
</tns:options>
<tns:attachments attachmentsParamName="attachmentsParamName__">
<tns:attachment>
<tns:attachmentID>attachmentID__</tns:attachmentID>
<tns:name>name_11__</tns:name>
<tns:localSourceLocation>localSourceLocation__</tns:localSourceLocation>
</tns:attachment>
</tns:attachments>
</tns:executeStartRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

<objLocation> especifica o local do formulário de solicitação inicial na biblioteca do CA Process Automation. Duas tags estão sob a tag <objLocation>.

<name> especifica o nome do formulário de solicitação inicial.

<path> fornece o caminho absoluto do objeto de formulário de solicitação inicial na biblioteca do CA Process Automation.

<tagId> Adiciona um ID exclusivo a um processo/formulário de solicitação inicial. Uma falha retorna se houver uma instância de processo ou de formulário de solicitação inicial com a mesma tag. A tag <tagId> monitora o status de qualquer processo/formulário de solicitação inicial para o qual você especificar a tag.

A tag <IsAutoArchive> pode ser definida como falso para ignorar instâncias de processo/formulário de solicitação inicial (juntamente com suas instâncias filho) do arquivamento na política de arquivamento. Processos que são executados em modo desvinculado não são afetados por essa tag. O valor padrão para <IsAutoArchive> é verdadeiro.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <executeStartRequestResponse xmlns="http://www.ca.com/itpam">
      <startRequestStatus>
        <interactionId>114</interactionId>
        <state>Running</state>
        <earliest-start-time>2012-03-01T00:00:00.000+05:30</earliest-start-time>
        <start-time>2012-03-01T13:54:13.523+05:30</start-time>
        <refProcess>/TENSU03/Process</refProcess>
        <params>
          <param name="DisableSchedulingDialog">True</param>
          <param name="FlowChartPath">/TENSU03/Process</param>
          <param name="userId">pamadmin</param>
          <param name="CALLER_TYPE">ITPAM_WS</param>
          <param name="Var_1">tentu</param>
          <param name="Var_0">sudhakar</param>
        </params>
      </startRequestStatus>
    </executeStartRequestResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

<interactionId> especifica a ID de interação da instância do formulário de solicitação inicial iniciado.

<state> mostra o estado da instância.

<refProcess> é o nome do processo com o caminho absoluto anexado ao formulário de solicitação inicial.

exportObject

Utilize este método para exportar uma biblioteca usando uma chamada de serviço web.

Exemplo

```
<tns:exportObject xmlns:tns="http://www.ca.com/itpam">
  <tns:auth>
    <!--xsd:Choice Type-->
    <tns:token>token__</tns:token>
    <tns:user>Joe</tns:user>
    <tns:password>thisismypassword</tns:password>
  </tns:auth>
  <tns:folderName>/SOAPTest/ObjectsToExport</tns:folderName>
  <tns:level>2</tns:level>
  <tns:isAbsolute>true</tns:isAbsolute>
  <tns:downloadFileName>
    <tns:downloadLocation>C:\</tns:downloadLocation>
    <tns:fileName>SOAPExportedObjects.xml</tns:fileName>
    <tns:overwriteFile>true</tns:overwriteFile>
  </tns:downloadFileName>
  <tns:filter>
    <tns:exportObjectTypes>
      <tns:objectType>Agenda</tns:objectType>
      <tns:objectType>Calendário</tns:objectType>
      <tns:objectType>Personalizar ícone</tns:objectType>
      <tns:objectType>Personalizar operador</tns:objectType>
      <tns:objectType>Conjunto de dados</tns:objectType>
      <tns:objectType>Formulário de solicitação de interação</tns:objectType>
      <tns:objectType>Pacote</tns:objectType>
      <tns:objectType>Processo</tns:objectType>
      <tns:objectType>Exibição de processos</tns:objectType>
      <tns:objectType>Recursos</tns:objectType>
      <tns:objectType>Formulário de solicitação inicial</tns:objectType>
    </tns:exportObjectTypes>
  </tns:filter>
</tns:exportObject>
```

Solicitação de SOAP

A solicitação contém:

- FolderName (a pasta ou o objeto que você deseja exportar)
- Pasta de nível de recursão
- Caminhos absolutos/relativos na saída de exportação
- Tipos de objeto
- Baixar o nome do arquivo e do local, onde a biblioteca exportada será salva.

Observação: todos os parâmetros são opcionais, com exceção de FolderName.

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:exportObject xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:folderName>/SOAPTest/ObjectsToExport</tns:folderName>
<tns:level>2</tns:level>
<tns:isAbsolute>true</tns:isAbsolute>
<tns:downloadFileName>
<tns:downloadLocation>C:\</tns:downloadLocation>
<tns:fileName>SOAPExportedObjects.xml</tns:fileName>
<tns:overwriteFile>true</tns:overwriteFile>
</tns:downloadFileName>
<tns:filter>
<tns:exportObjectTypes>
<tns:objectType>Agenda</tns:objectType>
<tns:objectType>Calendário</tns:objectType>
<tns:objectType>Personalizar ícone</tns:objectType>
<tns:objectType>Personalizar operador</tns:objectType>
<tns:objectType>Personalizar sensor</tns:objectType>
<tns:objectType>Conjunto de dados</tns:objectType>
<tns:objectType>Formulário de solicitação de interação</tns:objectType>
<tns:objectType>Pacote</tns:objectType>
<tns:objectType>Processo</tns:objectType>
<tns:objectType>Exibição de processos</tns:objectType>
<tns:objectType>Recursos</tns:objectType>
<tns:objectType>Formulário de solicitação inicial</tns:objectType>
<tns:objectType>Política de estado</tns:objectType>
<tns:objectType>Sistema</tns:objectType>
</tns:exportObjectTypes>
</tns:filter>
</tns:exportObject>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<ExportLibraryResponse xmlns="http://www.ca.com/itpam">
<exportLibraryResponse>Successfully uploaded the exported library to provided
download file name:C:\SOAPExportedObjects.xml</exportLibraryResponse>
<downloadedServer>PAM40-W2K8-17</downloadedServer>
</ExportLibraryResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

- Se você não fornecer o nome do arquivo de download na solicitação, a biblioteca serializada (objeto da biblioteca exportada) o enviará como um anexo para a resposta.
- Se você não fornecer um local compartilhado acessível e um nome de arquivo para a solicitação, a resposta de erro será retornada ao usuário.
- Se você fornecer um local compartilhado acessível e o Nome do arquivo de download, a biblioteca exportada será salva em um local determinado com o nome do arquivo especificado. O local compartilhado é retornado na resposta.
- Se Nome do arquivo de download que você fornecer já existir no local compartilhado, a resposta de erro será retornada ao usuário.

generateEvent

Use esse método para registrar um evento.

Exemplo

```
<tns:generateEvent xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:eventName>teste</tns:eventName>
<tns:eventType>eventType__</tns:eventType>
<tns:eventSource>testsource</tns:eventSource>
<tns:eventDestination>testdestination</tns:eventDestination>
<tns:eventExpDuration>130</tns:eventExpDuration>
<!--xsd:Choice Type-->
<tns:payload>teste</tns:payload>
<tns:params>
<tns:param name="name__">param__</tns:param>
</tns:params>
<tns:toSingleSubscriber>false</tns:toSingleSubscriber>
</tns:generateEvent>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<SOAP-ENV:Header/>

<SOAP-ENV:Body>

<tns:generateEvent xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:eventName>teste</tns:eventName>
<tns:eventType>eventType__</tns:eventType>
<tns:eventSource>testsource</tns:eventSource>
<tns:eventDestination>testdestination</tns:eventDestination>
<tns:eventExpDuration>130</tns:eventExpDuration>
<tns:payload>teste</tns:payload>
<tns:params>
<tns:param name="name__">param__</tns:param>
</tns:params>
<tns:toSingleSubscriber>false</tns:toSingleSubscriber>
</tns:generateEvent>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

<eventName> identifica o nome do evento (obrigatório).

<eventType> identifica o tipo de evento (opcional).

<eventSource> identifica a origem do evento (opcional).

<eventDestination> identifica o destino do evento (opcional).

<eventExpDuration> identifica o tempo de expiração do evento.

<payload> e <params> identificam dados de eventos adicionais. Se for um único valor, use payload; caso contrário, use params.

<toSingleSubscriber> identifica se o evento pode ser entregue a um ou vários assinantes.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<generateEventResponse xmlns="http://www.ca.com/itpam">
<event>
<eventId>b1a3c218-0a7d-4667-9b7c-9eeddcbc7408</eventId>
<eventName>test</eventName>
<eventType>eventType__</eventType>
<eventSource>testsource</eventSource>
<eventDestination>testdestination</eventDestination>
<payload>test</payload>
<toSingleSubscriber>false</toSingleSubscriber>
<eventCreationTime>2012-03-01T14:05:14.692+05:30</eventCreationTime>
<eventExpirationTime>2012-03-01T14:07:24.692+05:30</eventExpirationTime>
<user>pamadmin</user>
</event>
</generateEventResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

getAttachments

Use esse método para visualizar anexos.

Exemplo

```
<tns:getAttachments xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:filter>
<tns:contentId>111</tns:contentId>
<tns:contentType>text/html</tns:contentType>
</tns:filter>
</tns:getAttachments>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?><SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body><tns:getAttachments xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:filter>
<tns:contentId>111</tns:contentId>
<tns:contentType>text/html</tns:contentType>
</tns:filter>

</getAttachments></SOAP-ENV:Body></SOAP-ENV:Envelope>
```

A tag `<filter>` filtra a resposta com base na ID de conteúdo e no tipo de conteúdo do anexo (opcional).

`<contentId>` especifica a ID de conteúdo do anexo (opcional).

`<contentType>` especifica o tipo de conteúdo do anexo (opcional).

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<getAttachmentsResponse xmlns="http://www.ca.com/itpam">
<attachments>
<attachment>
<attachmentID>1</attachmentID>
<name>test</name>
<contentId>111</contentId>
<contentType>text/html</contentType>
</attachment>
</attachments>
</getAttachmentsResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Na resposta para cada anexo encontrado, há uma tag `<attachment>`.

Para cada tag de anexo, há as seguintes tags:

`<attachmentID>` especifica a ID do anexo.

`<name>` especifica o nome do anexo.

`<contentId>` especifica a ID de conteúdo do anexo.

`<contentType>` especifica o tipo de conteúdo do anexo.

getITPamVersionInfo

Use esse método para obter detalhes da versão do CA Process Automation.

Exemplo

```
<tns:getITPamVersion xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
</tns:getITPamVersion>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:getITPamVersion xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
</tns:getITPamVersion>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

`<servicePack>` identifica o service pack atual do CA Process Automation (opcional).

`<patchNumber>` identifica o número do patch atual do CA Process Automation (opcional).

`<buildDetails>` identifica todos os detalhes de criação que estão sendo usados com a versão atual do CA Process Automation (opcional).

`<majorVersion>` identifica a versão atual do CA Process Automation.

`<minorVersion>` identifica a versão secundária atual do CA Process Automation.

`<buildNumber>` identifica o número de criação atual do CA Process Automation.

`<buildDate>` identifica a data de criação atual do CA Process Automation.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <ITPamVersionDetails xmlns="http://www.ca.com/itpam">
      <majorVersion>4</majorVersion>
      <minorVersion>0</minorVersion>
      <patchNumber>0</patchNumber>
      <buildNumber>335</buildNumber>
      <buildDate>2012-03-01T06:08:09.000+05:30</buildDate>
    </ITPamVersionDetails>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

`<majorVersion>` identifica a versão atual do CA Process Automation.

`<minorVersion>` identifica a versão secundária atual do CA Process Automation.

`<patchNumber>` identifica o número do patch atual do CA Process Automation (opcional).

`<buildNumber>` identifica o número de criação atual do CA Process Automation.

`<buildDate>` identifica a data de criação atual do CA Process Automation.

`<buildDetails>` fornece mais detalhes sobre a criação (opcional).

getMatchingEvents

Use esse método para obter a lista de eventos correspondentes para um determinado assinante.

Exemplo

Allow partial Match True

```
<tns:getMatchingEvents xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:eventName>abc</tns:eventName>
<tns:eventType></tns:eventType>
<tns:eventSource></tns:eventSource>
<tns:eventDestination></tns:eventDestination>
<tns:eventExpression></tns:eventExpression>
<tns:enablePatternMatch></tns:enablePatternMatch>
<tns:allowPartialMatch>true</tns:allowPartialMatch>
<tns:consumeEvents></tns:consumeEvents>
<tns:retrieveAllMatchingEvents>true</tns:retrieveAllMatchingEvents>
<tns:uniqueId></tns:uniqueId>
</tns:getMatchingEvents>
```

Enable Pattern Match True

```
<tns:getMatchingEvents xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:eventName>a.*f</tns:eventName>
<tns:eventType></tns:eventType>
<tns:eventSource></tns:eventSource>
<tns:eventDestination></tns:eventDestination>
<tns:eventExpression></tns:eventExpression>
<tns:enablePatternMatch>true</tns:enablePatternMatch>
<tns:allowPartialMatch></tns:allowPartialMatch>
<tns:consumeEvents></tns:consumeEvents>
<tns:retrieveAllMatchingEvents>true</tns:retrieveAllMatchingEvents>
<tns:uniqueId></tns:uniqueId>
</tns:getMatchingEvents>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:getMatchingEvents xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:eventName>abc</tns:eventName>
<tns:eventType/>
<tns:eventSource/>
<tns:eventDestination/>
<tns:eventExpression/>
<tns:enablePatternMatch/>
<tns:allowPartialMatch/>
<tns:consumeEvents/>
<tns:retrieveAllMatchingEvents>true</tns:retrieveAllMatchingEvents>
<tns:uniqueId/>
</tns:getMatchingEvents>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

<eventName> identifica o nome do evento (opcional).

<eventType> identifica o tipo de evento (opcional).

<eventSource> identifica a origem do evento (opcional).

<eventDestination> identifica o destino do evento (opcional).

<eventExpression> é uma expressão booleana para parâmetros de eventos adicionais. Essa expressão é avaliada em relação aos parâmetros de evento ou carga que são enviados com um evento (opcional). Por exemplo:

```
Event.eventid==1234.
```

<enablePatternMatch> permite a correspondência de padrão para parâmetros como nome, tipo, origem e destino do evento.

<allowPartialMatch> permite uma correspondência parcial para parâmetros como nome, tipo, origem e destino do evento.

`<uniqueId>` envia uma ID exclusiva junto com a solicitação, de forma que a chamada SOAP consuma o evento apenas uma vez. Se o evento de envio tiver uma opção para entregar a vários assinantes, várias chamadas do `getMatchingEvents` com o mesmo `<uniqueId>` consumirão o evento apenas uma vez.

`<consumeEvents>` permite que a solicitação consuma ou não os eventos correspondentes.

`<retrieveAllMatchingEvents>` permite que a solicitação recupere o primeiro ou todos os eventos de envio correspondentes.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <getMatchingEventsResponse xmlns="http://www.ca.com/itpam">
      <events>
        <event>
          <eventId>fefce4c1-ad8e-42a2-8964-362704ddb24</eventId>
          <eventName>abc</eventName>
          <eventType>eventType__</eventType>
          <eventSource>eventSource__</eventSource>
          <eventDestination>eventDestination__</eventDestination>
          <payload>payload__</payload>
          <toSingleSubscriber>false</toSingleSubscriber>
          <eventCreationTime>2012-03-06T17:04:32.562+05:30</eventCreationTime>
          <eventExpirationTime>2012-03-06T17:06:42.562+05:30</eventExpirationTime>
          <user>pamadmin</user>
        </event>
      </events>
    </getMatchingEventsResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

getPendingInteractionRequestForm

Use esse método para recuperar as informações de um formulário de solicitação de interação pendente com base na ID da tarefa.

Exemplo

```
<tns:getPendingInteractionRequestForm xmlns:tns="http://www.ca.com/itpam"
getApprovalRequired="false" getParamSequence="false">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:userInteractionID>924</tns:userInteractionID>
</tns:getPendingInteractionRequestForm>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:getPendingInteractionRequestForm xmlns:tns="http://www.ca.com/itpam"
getApprovalRequired="false" getParamSequence="false">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:userInteractionID>924</tns:userInteractionID>
</tns:getPendingInteractionRequestForm>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

<userInteractionID> especifica a ID da tarefa do formulário de solicitação de interação pendente.

<getPendingInteractionRequestForm> recupera o número da sequência dos parâmetros e define o valor de atributo *getParamSequence* para "verdadeiro".

Resposta do SOAP

Com valores adequados:

```
<getPendingInteractionRequestFormResponse xmlns="http://www.ca.com/itpam">
  <processID>913</processID>
  <isApprovalRequired>true</isApprovalRequired>
  <description/>
  <pages hasComplexType="false">
    <page name="Parameters">
      <itpamString isReadOnly="false" maxLength="2147483647" minLength="0" name="Var_0"
        sequenceNo="0">
        <label>Var_0</label>
        <description/>
        <value/>
      </itpamString>
      <itpamString isReadOnly="false" maxLength="2147483647" minLength="0" name="Var_1"
        sequenceNo="1">
        <label>Var_1</label>
        <description/>
        <value/>
      </itpamString>
    </page>
    <page name="System"/>
    <page name="User Prompt"/>
  </pages>
</getPendingInteractionRequestFormResponse>
```

<pages> especifica os detalhes das páginas no formulário de solicitação de interação.

Quando uma tarefa é fornecida no estado Assumido:

```

<getPendingInteractionRequestFormResponse xmlns="http://www.ca.com/itpam">
<processID>7</processID>
<isApprovalRequired>true</isApprovalRequired>
<description/>
<pages hasComplexType="false">
<page name="Parameters">
<itpamString isReadOnly="false" maxLength="2147483647" minLength="0" name="Var_0"
sequenceNo="0">
<label>Var_0</label>
<description/>
<value/>
</itpamString>
<itpamString isReadOnly="false" maxLength="2147483647" minLength="0" name="Var_1"
sequenceNo="1">
<label>Var_1</label>
<description/>
<value/>
</itpamString>
</page>
<page name="System"/>
<page name="User Prompt"/>
</pages>
</getPendingInteractionRequestFormResponse>

```

Quando uma tarefa é fornecida no estado Concluído:

```

<Fault xmlns="http://schemas.xmlsoap.org/soap/envelope/">
<faultcode>SOAP-ENV:Server</faultcode>
<faultstring>"UserInteraction with Task Id "<<taskid>>" is in completed state
therefore this action is not allowed"
</faultstring>
</Fault>

```

Observação: a partir do CA Process Automation r2.2, "getApprovalRequired" é adicionado à solicitação "getPendingInteractionRequestForm". Se definido como verdadeiro, o sinalizador "isApprovalRequired" será retornado, indicando se o formulário precisa de aprovação. Para ativar esse comportamento nas instâncias do CA Process Automation que foram atualizadas a partir da r2.1, peça ao administrador do CA Process Automation para alterar a configuração do CA Process Automation (conforme descrito na Observação em "[Método executePendingInformation](#)" (na página 120)).

getPendingUserInteractions

Use este método para exibir as interações do usuário pendentes. O número de interações pendentes que retornam tem como base a função de usuário:

- Se o usuário for um administrador, todas as interações do usuário que estiverem pendentes no orquestrador retornam.
- No caso de qualquer outra função, somente as interações pendentes que foram atribuídas ao usuário específico retornam.

Retornar todas as interações que estão pendentes no orquestrador (apenas para administradores)

Texto embutido

```
<tns:getPendingUserInteractions xmlns:tns="http://www.ca.com/itpam"
getRootUUID="false">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:rootUUID></tns:rootUUID>
<tns:processID></tns:processID>
</tns:getPendingUserInteractions>
```

Solicitação

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:getPendingUserInteractions xmlns:tns="http://www.ca.com/itpam"
getRootUUID="false">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:rootUUID></tns:rootUUID>
<tns:processID></tns:processID>
</tns:getPendingUserInteractions>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Resposta

```
<getPendingUserInteractionsResponse xmlns="http://www.ca.com/itpam">
<userInteractions>
<userInteraction id="116" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="117" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="118" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="119" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="120" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="121" title="Task by B">
<description>Task by B</description>
</userInteraction>
```

Retornar todas as interações pendentes chamadas a partir de um formulário de solicitação inicial**Texto embutido**

Fornecer a rootUUID do processo que o formulário de solicitação inicial está chamando:

```
<tns:getPendingUserInteractions xmlns:tns="http://www.ca.com/itpam"
getRootUUID="false">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:rootUUID>076be822-7514-4f04-a333-3e8b73693a2b</tns:rootUUID>
<tns:processID></tns:processID>
</tns:getPendingUserInteractions>
```

Solicitação

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:getPendingUserInteractions xmlns:tns="http://www.ca.com/itpam"
getRootUUID="false">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:rootUUID>6e8faae6-77a4-4461-8feb-9e5ca4d572d5</tns:rootUUID>
<tns:processID></tns:processID>
</tns:getPendingUserInteractions>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Resposta

```
<getPendingUserInteractionsResponse xmlns="http://www.ca.com/itpam">
  <userInteractions>
    <userInteraction id="207" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
    <userInteraction id="208" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
    <userInteraction id="209" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
    <userInteraction id="210" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
    <userInteraction id="211" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
    <userInteraction id="212" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
    <userInteraction id="213" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
    <userInteraction id="214" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
    <userInteraction id="215" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
    <userInteraction id="216" title="Task by B">
      <description>Task by B</description>
    </userInteraction>
  </userInteractions>
</getPendingUserInteractionsResponse>
```

Processo pai chamando um processo filho vinculado

Texto embutido

Fornecer a rootUUID do processo pai a partir do conjunto de dados:

```
<tns:getPendingUserInteractions xmlns:tns="http://www.ca.com/itpam"
getRootUUID="false">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:rootUUID>6e8faae6-77a4-4461-8feb-9e5ca4d572d5</tns:rootUUID>
<tns:processID></tns:processID>
</tns:getPendingUserInteractions>
```

Solicitação

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:getPendingUserInteractions xmlns:tns="http://www.ca.com/itpam"
getRootUUID="false">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:rootUUID>6e8faae6-77a4-4461-8feb-9e5ca4d572d5</tns:rootUUID>
<tns:processID></tns:processID>
</tns:getPendingUserInteractions>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Resposta

```
<getPendingUserInteractionsResponse xmlns="http://www.ca.com/itpam">
<userInteractions>
<userInteraction id="176" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="177" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="178" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="179" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="180" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="181" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="182" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="183" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="184" title="Task by B">
<description>Task by B</description>
</userInteraction>
<userInteraction id="185" title="Task by B">
<description>Task by B</description>
</userInteraction>
</userInteractions>
</getPendingUserInteractionsResponse>
```

<userInteractions> especifica a ID e o título de cada interação do usuário pendente.

getProcessLogs

Use esse método para obter os logs de uma instância de processo.

Níveis de mensagem

Os logs de instância de processo do CA Process Automation exibem os seguintes níveis de mensagem:

- Erro =4
- Aviso =3
- Advertência =2
- Normal =1

Categorias de mensagem

As instâncias de processo do CA Process Automation têm as seguintes categorias de mensagem:

Processo

Registra as mensagens de uma instância de processo.

Operador

Registra as mensagens para os operadores em processo.

Manipulador

Registra as mensagens quando a alteração de rota ou o manipulador de exceções é chamado.

Resposta

Registra mensagens quando os agentes enviam uma resposta de volta ao orquestrador.

Personalizada

Quando mensagens são registradas sem nenhuma categoria, essas mensagens são agrupadas na categoria Personalizar. Você pode fornecer a categoria como Personalizada para recuperar essas mensagens.

Mensagens personalizadas

É possível definir suas próprias categorias de log. Por exemplo:

```
LogEvent(3, "SD_MESSAGES", "essa mensagem é registrada para operadores do Service Desk")
```

Para recuperar essas mensagens, especifique a categoria como SD_MESSAGES na solicitação.

Para recuperar as mensagens de log

Use a seguinte solicitação de SOAP:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:getProcessLogs xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:processID>72</tns:processID>
<tns:filter>
<tns:level>2</tns:level>
<tns:categories>
<tns:category>process</tns:category>
<tns:category>operator</tns:category>
<tns:category>handler</tns:category>
<tns:category>response</tns:category>
<tns:category> SD_MESSAGES </tns:category>
</tns:categories>
</tns:filter>
</tns:getProcessLogs>
</SOAP-ENV:Body></SOAP-ENV:Envelope>
```

`<level>` `</level>` pode ter um valor de erro, aviso, advertência e normal.

Se o usuário especificar o nível como “normal”, todas as mensagens com o nível normal e acima serão recuperadas.

Quando o erro estiver no nível mais elevado e o usuário especificar o nível como "erro", somente as mensagens com o nível "erro" serão recuperadas.

Se um usuário especificar um nível inválido, ele será tratado como normal (o menor nível).

Para recuperar logs com várias categorias:

```
<tns:categories>
  <tns:category>userMsg</tns:category>
  <tns:category>process</tns:category>
</tns:categories>
```

Nesse exemplo, todas as mensagens com a categoria “useMmsg” e “processo” são recuperadas. Somente as mensagens de log que corresponderem ao nível e a uma das categorias serão retornadas.

Exemplo

```
<tns:getProcessLogs xmlns:tns="http://www.ca.com/itpam">
  <tns:auth>
    <!--xsd:Choice Type-->
    <tns:user>Joe</tns:user>
    <tns:password>thisismypassword</tns:password>
  </tns:auth>
  <tns:processID>72</tns:processID>
  <tns:filter>
    <tns:level>2</tns:level>
    <tns:categories>
      <tns:category>process</tns:category>
      <tns:category>operator</tns:category>
      <tns:category>handler</tns:category>
      <tns:category>response</tns:category>
    </tns:categories>
  </tns:filter>
</tns:getProcessLogs>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <tns:getProcessLogs xmlns:tns="http://www.ca.com/itpam">
      <tns:auth>
        <tns:user>Joe</tns:user>
        <tns:password>thisismypassword</tns:password>
      </tns:auth>
      <tns:processID>72</tns:processID>
      <tns:filter>
        <tns:level>2</tns:level>
        <tns:categories>
          <tns:category>process</tns:category>
          <tns:category>operator</tns:category>
          <tns:category>handler</tns:category>
          <tns:category>response</tns:category>
        </tns:categories>
      </tns:filter>
    </tns:getProcessLogs>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <getProcessLogsResponse xmlns="http://www.ca.com/itpam">
      <processLogs>
        <processlog category="Process" level="Notice">
          <logMessage>'LaneChangeHandler_30_72' instance was created.</logMessage>
          <time>2012-03-01T13:18:43.248+05:30</time>
        </processlog>
        <processlog category="Process" level="Notice">
          <logMessage>'LaneChangeHandler_30_72' is in 'Queued' state.</logMessage>
          <time>2012-03-01T13:18:43.248+05:30</time>
        </processlog>
        <processlog category="Process" level="Notice">
          <logMessage>Process started at '03/01/2012 13:18:43' by 'padmin'.</logMessage>
          <time>2012-03-01T13:18:43.592+05:30</time>
        </processlog>
        <processlog category="Operator" level="Notice">
          <logMessage>'Start_Script_1' is enabled following 'Start_1'.</logMessage>
          <time>2012-03-01T13:18:43.639+05:30</time>
        </processlog>
        <processlog category="Operator" level="Notice">
          <logMessage>'Start_1' is 'Completed' on 'Current Server'.</logMessage>
          <time>2012-03-01T13:18:43.639+05:30</time>
        </processlog>
        <processlog category="Operator" level="Notice">
          <logMessage>A service request was sent for 'Start_Script_1'.</logMessage>
          <time>2012-03-01T13:18:44.123+05:30</time>
        </processlog>
        <processlog category="Operator" level="Notice">
          <logMessage>'Start_Script_1' is 'Running' on 'Current Server'.</logMessage>
          <time>2012-03-01T13:18:44.139+05:30</time>
        </processlog>
        <processlog category="Operator" level="Notice">
          <logMessage>'Start_Script_1' is 'Completed'.</logMessage>
          <time>2012-03-01T13:18:47.904+05:30</time>
        </processlog>
        <processlog category="Operator" level="Notice">
          <logMessage>'Start_Script_1_1' is enabled following 'Start_Script_1'.</logMessage>
          <time>2012-03-01T13:18:47.935+05:30</time>
        </processlog>
        <processlog category="Operator" level="Notice">
          <logMessage>'Lane_Change_1' was reset.</logMessage>
          <time>2012-03-01T13:18:47.982+05:30</time>
        </processlog>
        <processlog category="Operator" level="Notice">
          <logMessage>'Calculation_1' was reset.</logMessage>
          <time>2012-03-01T13:18:47.998+05:30</time>
        </processlog>
      </processLogs>
    </getProcessLogsResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_2' was reset.</logMessage>
<time>2012-03-01T13:18:48.014+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_1' was reset.</logMessage>
<time>2012-03-01T13:18:48.029+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_3' was reset.</logMessage>
<time>2012-03-01T13:18:48.029+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_2' was reset.</logMessage>
<time>2012-03-01T13:18:48.045+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1' is enabled following 'Lane_Change_1'.</logMessage>
<time>2012-03-01T13:18:48.045+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_1' is 'Completed' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:48.060+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>A service request was sent for 'Calculation_1'.</logMessage>
<time>2012-03-01T13:18:48.092+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1' is 'Running' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:48.092+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1' is 'Completed'.</logMessage>
<time>2012-03-01T13:18:48.576+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>A service request was sent for 'Start_Script_1_1'.</logMessage>
<time>2012-03-01T13:18:48.639+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Start_Script_1_1' is 'Running' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:48.654+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Start_Script_1_1' is 'Completed'.</logMessage>
<time>2012-03-01T13:18:49.060+05:30</time>
</processlog>
```

```
<processlog category="Operator" level="Notice">
<logMessage>'Start_Script_1_2' is enabled following
'Start_Script_1_1'.</logMessage>
<time>2012-03-01T13:18:49.076+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_1' was reset.</logMessage>
<time>2012-03-01T13:18:49.154+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1' was reset.</logMessage>
<time>2012-03-01T13:18:49.170+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_2' was reset.</logMessage>
<time>2012-03-01T13:18:49.201+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_1' was reset.</logMessage>
<time>2012-03-01T13:18:49.217+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_3' was reset.</logMessage>
<time>2012-03-01T13:18:49.217+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_2' was reset.</logMessage>
<time>2012-03-01T13:18:49.248+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_1' is enabled following 'Lane_Change_2'.</logMessage>
<time>2012-03-01T13:18:49.248+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_2' is 'Completed' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:49.248+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>A service request was sent for 'Calculation_1_1'.</logMessage>
<time>2012-03-01T13:18:49.279+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_1' is 'Running' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:49.295+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_1' is 'Completed'.</logMessage>
<time>2012-03-01T13:18:49.639+05:30</time>
</processlog>
```

```
<processlog category="Operator" level="Notice">
<logMessage>A service request was sent for 'Start_Script_1_2'.</logMessage>
<time>2012-03-01T13:18:49.701+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Start_Script_1_2' is 'Running' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:49.701+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Start_Script_1_2' is 'Completed'</logMessage>
<time>2012-03-01T13:18:50.123+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Start_Script_1_3' is enabled following
'Start_Script_1_2'.</logMessage>
<time>2012-03-01T13:18:50.154+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_1' was reset.</logMessage>
<time>2012-03-01T13:18:50.217+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1' was reset.</logMessage>
<time>2012-03-01T13:18:50.248+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_2' was reset.</logMessage>
<time>2012-03-01T13:18:50.264+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_1' was reset.</logMessage>
<time>2012-03-01T13:18:50.279+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_3' was reset.</logMessage>
<time>2012-03-01T13:18:50.310+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_2' was reset.</logMessage>
<time>2012-03-01T13:18:50.326+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_2' is enabled following 'Lane_Change_3'.</logMessage>
<time>2012-03-01T13:18:50.342+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Lane_Change_3' is 'Completed' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:50.342+05:30</time>
</processlog>
```

```
<processlog category="Operator" level="Notice">
<logMessage>A service request was sent for 'Calculation_1_2'.</logMessage>
<time>2012-03-01T13:18:50.373+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_2' is 'Running' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:50.373+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Calculation_1_2' is 'Completed'</logMessage>
<time>2012-03-01T13:18:50.842+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>A service request was sent for 'Start_Script_1_3'.</logMessage>
<time>2012-03-01T13:18:51.107+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Start_Script_1_3' is 'Running' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:51.123+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Start_Script_1_3' is 'Completed'</logMessage>
<time>2012-03-01T13:18:55.982+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Normal_Stop_1' is enabled following 'Start_Script_1_3'.</logMessage>
<time>2012-03-01T13:18:56.014+05:30</time>
</processlog>
<processlog category="Operator" level="Notice">
<logMessage>'Normal_Stop_1' is 'Completed' on 'Current Server'.</logMessage>
<time>2012-03-01T13:18:56.060+05:30</time>
</processlog>
<processlog category="Process" level="Notice">
<logMessage>Process is 'Completed'.</logMessage>
<time>2012-03-01T13:18:56.060+05:30</time>
</processlog>
</processLogs>
</getProcessLogsResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

getProcessStatus

Use este método para visualizar o estado de uma instância do processo.

Exemplo

```
<tns:getProcessStatus xmlns:tns="http://www.ca.com/itpam">
<tns:flow>
<!--xsd:Choice Type-->
<tns:ROID>489</tns:ROID>
<tns:action>verificar</tns:action>
<tns:auth>
<!--xsd:Choice Type-->
<tns:token>token__</tns:token>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
</tns:flow>
</tns:getProcessStatus>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:getProcessStatus xmlns:tns="http://www.ca.com/itpam">
<tns:flow>
<tns:ROID>489</tns:ROID>
<tns:tagId>z34dsf5c7-xagb-4g5d-74g5-bd5c4we2378f</tns:tagId>
<tns:action>verificar</tns:action>
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
</tns:flow>
</tns:getProcessStatus>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Você pode incluir um dos itens a seguir:

- `<ROID>` especifica a ROID da instância de processo que você está exibindo o estado do sistema.
- `<tagId>`: o usuário passa a tagId do formulário de solicitação inicial para exibição.

`<action>` especifica a ação a ser executada na instância especificada; o valor desta tag deve ser verificado.

Respostas do SOAP

Processo concluído

```
<CheckFlowResponse xmlns="http://www.ca.com/itpam">
<CheckFlowResult>
<ROID>489</ROID>
<flow-state>Finalizado</flow-state>
<submission-time>2012-02-01T20:15:41.515+05:30</submission-time>
<sender-address>nenhum</sender-address>
<connection-id>nenhum</connection-id>
<message>Estado do fluxo recebido</message>
<params>
<param name="RuntimeROID">489</param>
<param name="Result">1</param>
<param name="ServerId">b9c4b8de-b74b-44b5-b2d2-1deb6e3dc872</param>
<param name="effectiveUser">pamadmin</param>
<param name="InstanceUUID">f120811e-3895-41a2-abbb-3458c19a51c8</param>
<param name="UserName">pamadmin</param>
<param name="ProcessName">/SOAPTest/P1</param>
<param name="ServerName">lodivsa204.ca.com</param>
<param name="DisplayName">P1</param>
<param name="TouchpointName">Orquestrador</param>
<param name="CallerUser">pamadmin</param>
<param name="EndDate">2012-02-01 16:30:24</param>
<param name="StartDate">2012-02-01 16:30:24</param>
<param name="ScheduledStartTime">2012-02-01 00:00:00</param>
<param name="rootUUID">f120811e-3895-41a2-abbb-3458c19a51c8</param>
<param name="EndTime">2012-02-01 16:30:24</param>
<param name="StartTime">2012-02-01 16:30:24</param>
<param name="ProcessAction">start</param>
<param name="InstanceName">P1_489</param>
</params>
</CheckFlowResult>
</CheckFlowResponse>
```

<flow-state> retorna o estado da instância do processo.

Processo inexistente ou arquivado

O operador falha com um problema e retorna esta resposta:

```
<Fault xmlns="http://schemas.xmlsoap.org/soap/envelope/">
<faultcode>SOAP-ENV:Server</faultcode>
<faultstring>The instance does not exist. It may have been archived. Refresh the
screen.</faultstring>
</Fault>
```

Ação inválida

O operador falha com este problema:

```
<Fault xmlns="http://schemas.xmlsoap.org/soap/envelope/">  
<faultcode>SOAP-ENV:Server</faultcode>  
<faultstring>Invalid flow action: c</faultstring>  
</Fault>
```

ROID inválida

O operador falha com este problema:

```
<Fault xmlns="http://schemas.xmlsoap.org/soap/envelope/">  
<faultcode>SOAP-ENV:Server</faultcode>  
<faultstring>error retrieving flow state</faultstring>  
</Fault>
```

getStartRequestForm

Use esse método para recuperar informações sobre um formulário de solicitação inicial que se baseia no nome e no local do formulário de solicitação inicial da biblioteca do CA Process Automation.

Exemplo

```
<tns:getStartRequestForm xmlns:tns="http://www.ca.com/itpam"  
getParamSequence="true">  
<tns:auth>  
<!--xsd:Choice Type-->  
<tns:token>token__</tns:token>  
<tns:user>Joe</tns:user>  
<tns:password>thisismypassword</tns:password>  
<tns:auth>  
<tns:objLocation>  
<tns:name>MYSRF</tns:name>  
<tns:path>/myfolder/</tns:path>  
</tns:objLocation>  
</tns:getStartRequestForm>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/><SOAP-ENV:Body>
<tns:getStartRequestForm xmlns:tns="http://www.ca.com/itpam"
getParamSequence="true">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:objLocation>
<tns:name>MYSRF</tns:name>
<tns:path>/TENSU03</tns:path>
</tns:objLocation>
</tns:getStartRequestForm>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

`<name>` é o nome do formulário de solicitação inicial.

`<path>` é o caminho absoluto da pasta em que reside o formulário de solicitação inicial.

Resposta do SOAP

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header/>
  <SOAP-ENV:Body>
    <getStartRequestFormResponse xmlns="http://www.ca.com/itpam">
      <refProcess>/Joe/Process</refProcess>
      <pages hasComplexType="false">
        <page name="Page">
          <itpamString isReadOnly="false" maxLength="2147483647" minLength="0" name="Var_0"
            sequenceNo="0">
            <label>Text Field</label>
            <description/>
            <value/>
          </itpamString>
          <itpamString isReadOnly="false" maxLength="2147483647" minLength="0" name="Var_1"
            sequenceNo="1">
            <label>Text Field</label>
            <description/>
            <value/>
          </itpamString>
        </page>
        <page name="System"/>
      </pages>
    </getStartRequestFormResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

<refProcess> mostra o nome e o caminho do processo anexado a este formulário de solicitação inicial.

getStartRequestForms

Use esse método para pesquisar os formulários de solicitação inicial na biblioteca do CA Process Automation. Uma pesquisa é baseada no caminho e na palavra-chave fornecida.

Exemplo

Com /root como o caminho de pesquisa

```
<tns:getStartRequestForms xmlns:tns="http://www.ca.com/itpam">
  <tns:auth>
    <!--xsd:Choice Type-->
    <tns:token>token__</tns:token>
    <tns:user>Joe</tns:user>
    <tns:password>thisismypassword</tns:password>
  </tns:auth>
  <tns:filter>
    <tns:lookUpPath isRecursive="true"></tns:lookUpPath>
    <tns:keywords>
      <tns:keyword>ABC</tns:keyword>
    </tns:keywords>
  </tns:filter>
</tns:getStartRequestForms>
```

Para obter todos os formulários de solicitação inicial no ambiente

Forneça / e remova Keyword__ das tags Keyword:

```
<tns:getStartRequestFormsResponse xmlns:tns="http://www.ca.com/itpam">
<startRequests>
<tns:startRequest name="Formulário de solicitação de compra inicial" refPath="/PAM
Hardware Procurement/">
<tns:description/>
</tns:startRequest>
<tns:startRequest name="Asset_Install_SW_SRF" refPath="/sinral/CA ITAM/SRF/">
<tns:description>Objeto do banco de dados</tns:description>
</tns:startRequest>
<tns:startRequest name="Asset_MAC_HW_SRF" refPath="/sinral/CA ITAM/SRF/">
<tns:description>Objeto do banco de dados</tns:description>
</tns:startRequest>
<tns:startRequest name="Asset_Transfer_SRF" refPath="/sinral/CA ITAM/SRF/">
<tns:description>Objeto do banco de dados</tns:description>
</tns:startRequest>
<tns:startRequest name="Asset_Disposal_HW_SRF" refPath="/sinral/CA ITAM/SRF/">
<tns:description>Objeto do banco de dados</tns:description>
</tns:startRequest>
<tns:startRequest name="Asset_Install_HW_Transferred_SRF" refPath="/sinral/CA
ITAM/SRF/">
<tns:description>Objeto do banco de dados</tns:description>
</tns:startRequest>
<tns:startRequest name="Asset_Install_HW_SRF" refPath="/sinral/CA ITAM/SRF/">
<tns:description>Objeto do banco de dados</tns:description>
</startRequest>
<tns:startRequest name="-Formulário de solicitação inicial" refPath="/Package
Demo/">
<tns:description/>
</tns:startRequest>
<tns:startRequest name="-Formulário de solicitação inicial" refPath="/Jack/">
<tns:description/>
</tns:startRequest>
</tns:startRequests>
</tns:getStartRequestFormsResponse>
```

Executar uma pesquisa de palavra-chave

Você pode pesquisar formulários de solicitação inicial com palavras-chave que estão associadas a eles. Por exemplo, você pode executar uma pesquisa para verificar se *SRF1* e *SRF2* estão associados à palavra-chave "ABC" na pasta */myfolder*.

```
<tns:getStartRequestForms xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:filter>
<tns:lookUpPath isRecursive="true"/>myfolder</tns:lookUpPath>
<tns:keywords>
<tns:keyword>ABC</tns:keyword>
</tns:keywords>
</tns:filter>
</tns:getStartRequestForms>
```

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:getStartRequestForms xmlns:tns="http://www.ca.com/itpam">
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:filter>
<tns:lookUpPath isRecursive="true"/></tns:lookUpPath>
<tns:keywords>
<tns:keyword>ABC</tns:keyword>
</tns:keywords>
</tns:filter>
</tns:getStartRequestForms>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Esta tag está presente na tag `<filter>`:

- `<lookUpPath>` fornece o caminho em que o usuário deseja pesquisar os formulários de solicitação inicial. Um atributo nesta tag é recorrente. Se um usuário desejar pesquisar de maneira recorrente no caminho fornecido, defina este atributo como verdadeiro. Se um usuário desejar pesquisar apenas no caminho fornecido, defina este atributo como falso. Forneça o caminho completo (como `/Folder/Folder1`) como o valor dessa tag na qual o usuário deseja pesquisar.

Resposta do SOAP

```
<getStartRequestFormsResponse xmlns="http://www.ca.com/itpam">
  <startRequests>
    <startRequest name="SRF1" refPath="/TENSU03/">
      <description/>
    </startRequest>
    <startRequest name="SRF2" refPath="/TENSU03/">
      <description/>
    </startRequest>
  </startRequests>
</getStartRequestFormsResponse>
```

Para cada formulário de solicitação inicial que atender aos critérios da pesquisa, haverá um `<startRequest>` sob a tag `<startRequests>`. Esta tag contém o nome, o caminho de referência e a descrição do formulário de solicitação inicial na resposta de SOAP.

ImportObject

Use esse método para importar objetos para uma biblioteca do CA Process Automation.

Exemplo

```
<tns:ImportObject xmlns:tns="http://www.ca.com/itpam">
  <tns:importFileParam>
    <tns:auth>
      <!--xsd:Choice Type-->
      <tns:token>token__</tns:token>
      <tns:user>Joe</tns:user>
      <tns:password>thisismypassword</tns:password>
    </tns:auth>
    <tns:importLocation>importLocation__</tns:importLocation>
    <tns:localSourceLocation>localSourceLocation__</tns:localSourceLocation>
    <tns:isSetCurrent>isSetCurrent__</tns:isSetCurrent>
    <tns:isMakeAvailable>isMakeAvailable__</tns:isMakeAvailable>
    <tns:overwriteAction>overwriteAction__</tns:overwriteAction>
  </tns:importFileParam>
  <!--This SOAP request can have MIME contents-->
</tns:ImportObject>
```

`<isSetCurrent>` especifica se os objetos importados devem ser marcados como versões atuais (opcional).

`<isMakeAvailable>` especifica se operadores personalizados importados devem ser marcados como disponíveis (opcional).

Depois de fornecer valores no texto embutido, crie um anexo MIME para o arquivo que você especificou na tag `<localSourceLocation>` da solicitação de SOAP. Para adicionar um anexo MIME, no painel Propriedades, em Anexos de MIME, clique em Adicionar e, em seguida, forneça os valores apropriados para:

Tipo de conteúdo

Tipo de conteúdo que o anexo MIME está carregando. Por exemplo, text/xml.

ID do conteúdo

ID que com a qual o anexo é identificado de forma exclusiva. Por exemplo, 11111.

URL do arquivo

Caminho do anexo. Esse caminho pode ser o caminho do arquivo .xml a ser importado. Por exemplo, C:\\SOAPExportedObjects.xml.

Salve e execute o processo.

Solicitação de SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<tns:ImportObject xmlns:tns="http://www.ca.com/itpam">
<tns:importFileParam>
<tns:auth>
<tns:user>Joe</tns:user>
<tns:password>thisismypassword</tns:password>
<tns:auth>
<tns:importLocation></tns:importLocation>
<tns:localSourceLocation>"C:\\SOAPExportedObjects.xml"</tns:localSourceLocation>
<tns:isSetCurrent>true</tns:isSetCurrent>
<tns:isMakeAvailable>true</tns:isMakeAvailable>
<tns:overwriteAction>incrementObjectVersion</tns:overwriteAction>
</tns:importFileParam>
<!--This request can have MIME contents-->
</tns:ImportObject>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

`<importLocation>` especifica o local na biblioteca do CA Process Automation onde você deseja importar os objetos.

`<isSetCurrent>` especifica se os objetos importados devem ser marcados como versões atuais (opcional). O valor padrão é falso.

`<isMakeAvailable>` especifica se operadores personalizados importados devem ser marcados como disponíveis (opcional). O valor padrão é falso.

`<overwriteAction>`

Se já existir um objeto com o mesmo nome na biblioteca, você poderá selecionar uma das seguintes opções para `<overwriteAction>`:

- `incrementObjectVersion` - Importar como uma nova versão e manter o objeto existente.
- `replaceObject` - Importar e substituir o objeto existente.
- `skipImport` - Não importar objetos com o mesmo nome como um objeto existente.

Por exemplo:

```
<tns:ImportObject xmlns:tns="http://www.ca.com/itpam">
  <tns:importFileParam>
    <tns:auth>
      <tns:user>Joe</tns:user>
      <tns:password>thisismypassword</tns:password>
    </tns:auth>
    <tns:importLocation>/ABC</tns:importLocation>
    <tns:localSourceLocation>"C://abc.xml"</tns:localSourceLocation>
    <tns:isSetCurrent>true</tns:isSetCurrent>
    <tns:isMakeAvailable>true</tns:isMakeAvailable>
    <tns:overwriteAction>skipImport</tns:overwriteAction>
  </tns:importFileParam>
  <!-- This SOAP request can have MIME contents-->
</tns:ImportObject>
```

Resultado:

```
<importFileResponse xmlns="http://www.ca.com/itpam">
  <messages>
    <message>Não foi possível importar o objeto CustomIcon. Outro objeto com o mesmo nome "CustomIcon" já existe.</message>
    <message>Não foi possível importar o objeto P1. Outro objeto com o mesmo nome "P1" já existe.</message>
    <message>Não foi possível importar o objeto Processo. Outro objeto com o mesmo nome "Processo" já existe.</message>
    <message>Não foi possível importar o objeto Agenda. Outro objeto com o mesmo nome "Agenda" já existe.</message>
    <message>Não foi possível importar o objeto Recursos. Outro objeto com o mesmo nome "Recursos" já existe.</message>
    <message>Não foi possível importar o objeto SOAPMethods. Outro objeto com o mesmo nome "SOAPMethods" já existe.</message>
    <message>Não foi possível importar o objeto ProcessWatch. Outro objeto com o mesmo nome "ProcessWatch" já existe.</message>
    <message>Não foi possível importar o objeto Calendário. Outro objeto com o mesmo nome "Calendário" já existe.</message>
  </messages>
</importFileResponse>
```

```

<message>Não foi possível importar o objeto SOAPModule_NegativeCases. Outro
objeto com o mesmo nome "SOAPModule_NegativeCases" já existe.</message>
<message>Não foi possível importar o objeto IRF. Outro objeto com o mesmo nome
"IRF" já existe.</message>
<message>Não foi possível importar o objeto SOAPOperators_DynamicParams.
Outro objeto com o mesmo nome "SOAPOperators_DynamicParams" já
existe.</message>
<message>Não foi possível importar o objeto Formulário de solicitação de
interação. Outro objeto com o mesmo nome "Formulário de solicitação de
interação" já existe.</message>
<message>Não foi possível importar o objeto SRF. Outro objeto com o mesmo nome
"SRF" já existe.</message>
<message>Não foi possível importar o objeto SOAPTest_Trigger. Outro objeto
com o mesmo nome "SOAPTest_Trigger" já existe.</message>
<message>Não foi possível importar o objeto CustomOperator. Outro objeto com
o mesmo nome "CustomOperator" já existe.</message>
<message>Não foi possível importar o objeto Process_2. Outro objeto com o
mesmo nome "Process_2" já existe.</message>
<message>Não foi possível importar o objeto Conjunto de dados. Outro objeto
com o mesmo nome "Conjunto de dados" já existe.</message>
<message>Não foi possível importar o objeto Formulário de solicitação
inicial. Outro objeto com o mesmo nome "Formulário de solicitação inicial"
já existe.</message>
<message>Não foi possível importar o objeto GlobalVars. Outro objeto com o
mesmo nome "GlobalVars" já existe.</message>
<message>Não foi possível importar o objeto Pacote. Outro objeto com o mesmo
nome "Pacote" já existe.</message>
</messages>
</importFileResponse>

```

Resposta do SOAP

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<importFileResponse xmlns="http://www.ca.com/itpam">
<successMessage>The specified object imported successfully.</successMessage>
</importFileResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

<successMessage> exibe uma mensagem de importação com êxito.

Tags comuns para métodos de serviços web

As seguintes tags comuns aplicam-se a vários métodos de serviço web:

A tag <pages> (na página 173)

Esta tag é usada nos seguintes serviços web:

- [Método `getPendingInteractionRequestForm`](#) (na página 142)
- [Método `getStartRequestForm`](#) (na página 162)

A tag <attachments> (na página 175)

Esta tag é usada nos seguintes serviços web:

- [Método `deleteAttachments`](#) (na página 119)
- [Método `executeStartRequest`](#) (na página 126)
- [Método `getAttachments`](#) (na página 136)
- [Método `excuteProcess`](#) (na página 123)

A tag <params> (na página 176)

Esta tag é usada nos seguintes serviços web:

- [Método `checkStartRequestStatus`](#) (na página 111)
- [Método `executePendingInformation`](#) (na página 120)
- [Método `executeProcess`](#) (na página 123)
- [Método `executeStartRequest`](#) (na página 126)
- [Método `getProcessStatus`](#) (na página 159)

A tag <options> (na página 176)

Esta tag é usada nos seguintes serviços web:

- [Método `executeProcess`](#) (na página 123)
- [Método `executeStartRequest`](#) (na página 126)

As tags <page> e <pages>

Em uma resposta do SOAP, a tag <pages> representa as páginas em um formulário de solicitação inicial/formulário de solicitação de interação. Para cada página no formulário de solicitação inicial/formulário de solicitação de interação, há uma tag <page> na tag <pages>.

Os parâmetros dessa página são exibidos sob a tag <page>. A tag <page> contém um nome de atributo que exibe o nome da página.

Há uma tag correspondente em <Page> para a maioria dos tipos de variáveis no conjunto de dados de formulário.

Mapeamento entre variáveis do conjunto de dados e tags em <Page>.

- Boolean\itpamBoolean
- Date\itpamDate
- Double\itpamDouble
- Integer\itpamInt
- Long\itpamLong
- Password\itpamPassword
- String\itpamString

Observação: a CA Technologies não oferece suporte a MapasDeValor, referências de objeto ou matriz por meio de serviços Web.

Os atributos do elemento de parâmetro incluem:

- Atributos de validação do parâmetro
- Nome do parâmetro

O seguinte é exibido neste nó:

- Rótulo
- Valor (não retornado em caso de senha)
- Descrição (retornado se não estiver em branco)
- Valor padrão (retornado se não estiver em branco)

A tag <option> é exibida com as opções que você especifica (se especificar os valores predefinidos para um parâmetro em um formulário de solicitação inicial/formulário de solicitação de interação). O atributo isallowOtherValue da tag <option> é verdadeiro se o usuário puder especificar um valor diferente do fornecido na opção.

Exemplo

```
<pages hasComplexType="false">
  <page name="Parameters">
    <itpamString isReadOnly="false" maxLength="254" minLength="0"
name="Var_0" sequenceNo="0">
      <label>Var_0</label>
      <description/>
      <value>pink</value>
    <defaultValue>op10</defaultValue>
    <options isallowOtherValue="false">
      <op name="option1">op1</op>
      <op name=" option2">op2</op>
    </options>
  </itpamString>
  <itpamInt isReadOnly="false" maxval="2147483647"
minval="-2147483648" name="Var_4" sequenceNo="4">
    <label>Var_4</label>
    <description/>
    <value>0</value>
    <defaultValue>0</defaultValue>
  </itpamInt>
  <itpamLong isReadOnly="false" maxval="9223372036854775807"
minval="-9223372036854775808" name="Var_1" sequenceNo="1">
    <label>Var_1</label>
    <description/>
    <value>0</value>
    <defaultValue>0</defaultValue>
  </itpamLong>
  <itpamDouble isReadOnly="false" maxval="1.7976931348623157E308"
minval="-1.7976931348623157E308" name="Var_2" sequenceNo="2">
```

```

        <label>Var_2</label>
        <description/>
        <value>0,0</value>
        <defaultValue>0,0</defaultValue>
    </itpamDouble>
    <itpamDate isReadOnly="false" name="Var_3" sequenceNo="3">
        <label>Var_3</label>
        <description/>
    </itpamDate>
    <itpamBoolean isReadOnly="false" name="Var_5" sequenceNo="5">
        <label>Var_5</label>
        <description/>
        <value>falso</value>
        <defaultValue>falso</defaultValue>
    </itpamBoolean>
</page>
<page name="User Prompt"/>
<page name="System"/>
</pages>          <page name="System"/>
</pages>

```

A tag <attachments>

A tag <attachments> fornece informações meta sobre o anexo. O CA Process Automation oferece suporte à adição de vários anexos para os métodos executeProcess e executeStartRequestForm.

Exemplo:

```

<attachments attachmentsParamName="?">
  <attachment
    <attachmentID> </attachmentID>
    <name>?</name>
  </attachment>

```

O <attachmentID> fornece a ID do conteúdo do anexo.

O <name> especifica o nome do anexo.

O <attachmentsParamName = "paramName"> especifica o nome da variável que é exibida no conjunto de dados do processo da instância de processo executada. O anexo pode ser acessado usando a variável do nome *paramName*.

Importante: Na integração ao CA Process Automation usando serviços web, os stubs do serviço web gerados usando ferramentas comuns de terceiros (como wsdl2java) não geram claramente o código para manipulação de anexos. Para integração no CA Process Automation usando os serviços web que manipulam anexos, consulte a seção "Utilitários web do CA IT PAM" da página "Práticas recomendadas do CA IT PAM". Essa página está vinculada a partir da página inicial do CA IT PAM em support.ca.com.

A tag <params>

Para fornecer um parâmetro, crie uma tag <params>. Você pode fornecer uma lista de parâmetros na tag <params> na solicitação.

Exemplo:

```
<params>
  <param name="vendorName">CA </param>
  <param name="quantity"> 4</param>
  <param name="start Date">30/10/2000</param>
  <param name="approvalRequired">false</param>
</params>
```

O atributo name especifica o nome do parâmetro.

Value especifica o valor do parâmetro no atributo de valor.

- Se o parâmetro já existir no conjunto de dados do objeto do CA Process Automation, o valor do parâmetro será ajustado com o novo valor.
- Se o parâmetro não existir, uma nova variável é criada com o nome e o valor fornecido. O tipo de variável depende do valor fornecido.

A tag <options>

A tag <options> contém os parâmetros de programação.

Exemplo

```
<options>
  <startDate> 2002-09-24</startDate>
  <startTime> 09:00:00 </startTime>
</options>
```

O processo é executado de acordo com a data de início e a hora de início fornecidas.

Apêndice A: Status do HTTP e códigos de erro

Cada resposta HTTP contém um código de status de HTTP. A numeração dos códigos de respostas HTTP bem-sucedidas varia de 200 a 399. A numeração padrão dos códigos de respostas HTTP varia de 400 a 599.

A lista a seguir mostra os números de códigos com êxito e códigos de erro que a API do REST retorna para cada um dos métodos HTTP.

GET (única)

200, 400, 401, 404, 409

GET (conjunto)

200, 400, 401

PUT

200, 400, 401, 404, 409

DELETE

204, 400, 401, 409

POST

201, 400, 401, 409

Observação: em relação às operações de PUT e DELETE, a API *não* tentará determinar, inicialmente, a validade da ID. Em vez disso, o API tentará executar a atualização e excluir as consultas diretamente. Se ocorrer um erro, a API retorna o código de *Conflito 409*. Se nessa situação a API retornar um código *404 Não encontrado*, o desempenho será afetado.

Além disso, as seguintes mensagens de erro de retornam sob os casos abaixo:

- Se a solicitação HTTP contiver um endereço de URI inválido ou inacessível, o servidor responderá com o código de resposta *404 - Não Encontrado*.
- Se a solicitação de HTTP não contiver um método HTTP sem suporte para uma URI válida, o servidor responderá com o código de resposta *405 - Método Não Permitido*.

- Se a solicitação HTTP pedir um tipo de mídia para o qual não há suporte (Aceitar cabeçalho), o servidor responde com o código de resposta *406 - Não Aceitável*.
- Se a solicitação de HTTP envia um tipo de mídia para o qual não há suporte (cabeçalho do tipo de conteúdo), o servidor responde com o código de resposta *415 Tipo de Mídia Não Suportado*.
- Vários erros de sintaxe ou internos do Servidor web podem retornar um erro interno 500.

Status do HTTP e códigos de erro

Cada resposta HTTP contém um código de status de HTTP. A numeração dos códigos de respostas HTTP bem-sucedidas varia de 200 a 399. A numeração padrão dos códigos de respostas HTTP varia de 400 a 599.

A lista a seguir mostra os números de códigos com êxito e códigos de erro que a API do REST retorna para cada um dos métodos HTTP.

GET (única)

200, 400, 401, 404, 409

GET (conjunto)

200, 400, 401

PUT

200, 400, 401, 404, 409

DELETE

204, 400, 401, 409

POST

201, 400, 401, 409

Observação: em relação às operações de PUT e DELETE, a API *não* tentará determinar, inicialmente, a validade da ID. Em vez disso, o API tentará executar a atualização e excluir as consultas diretamente. Se ocorrer um erro, a API retorna o código de *Conflito 409*. Se nessa situação a API retornar um código *404 Não encontrado*, o desempenho será afetado.

Além disso, as seguintes mensagens de erro de retornam sob os casos abaixo:

- Se a solicitação HTTP contiver um endereço de URI inválido ou inacessível, o servidor responderá com o código de resposta *404 - Não Encontrado*.
- Se a solicitação de HTTP não contiver um método HTTP sem suporte para uma URI válida, o servidor responderá com o código de resposta *405 - Método Não Permitido*.

- Se a solicitação HTTP pedir um tipo de mídia para o qual não há suporte (Aceitar cabeçalho), o servidor responde com o código de resposta *406 - Não Aceitável*.
- Se a solicitação de HTTP envia um tipo de mídia para o qual não há suporte (cabeçalho do tipo de conteúdo), o servidor responde com o código de resposta *415 Tipo de Mídia Não Suportado*.
- Vários erros de sintaxe ou internos do Servidor web podem retornar um erro interno 500.

Códigos de Status conhecidos

A lista a seguir descreve os códigos de status conhecidos que a API retorna. Outros códigos podem existir no servidor web ou no framework CXF estrutura, mas depende do tipo de erro.

200

OK

Indica um retorno bem-sucedido.

201

Criado

Indica um novo registro.

204

Sem conteúdo

Indica um corpo de resposta vazio.

304

Não modificado

Indica que o registro foi atualizado.

400

Solicitação incorreta

Indica que ocorreu um erro devido a uma ocorrência do usuário ou do servidor de back-end.

401

Não autorizado

Indica que um erro de acesso à função ou falha de autenticação.

404

Não encontrado

Indica que um registro não foi encontrado.

405

Método não permitido

Indica um método HTTP para o qual não há suporte.

406

Não aceitável

Indica que não há suporte para o formato requisitado.

409

Conflito

Indica que vários registros foram encontrados para o identificador fornecido.

415

Tipo de mídia não suportado

Indica que o formato fornecido não é suportado.

500

Erro interno do servidor

Indica um erro no servidor ou framework CXF.